# NEAR-OPTIMALITY OF DISTRIBUTED NETWORK MANAGEMENT WITH A MACHINE LEARNING APPROACH

A Thesis
Presented to
The Academic Faculty

by

Sung-eok Jeon

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
School of Electrical and Computer Engineering

Georgia Institute of Technology
August 2007

# NEAR-OPTIMALITY OF DISTRIBUTED NETWORK MANAGEMENT WITH A MACHINE LEARNING APPROACH

Approved by:


Professor Chuanyi Ji, Advisor
School of Electrical and Computer Engineering
*Georgia Institute of Technology*

Professor Steven W. McLaughlin
School of Electrical and Computer Engineering
*Georgia Institute of Technology*

Professor Raghupathy Sivakumar
School of Electrical and Computer Engineering
*Georgia Institute of Technology*

Professor Mustafa Ammar
College of Computing
*Georgia Institute of Technology*

Professor John R. Barry
School of Electrical and Computer Engineering
*Georgia Institute of Technology*


Date Approved: 6 July 2007

*For Parents.*

# ACKNOWLEDGEMENTS

I would like to express my sincere gratitude to my advisor, Professor Chuanyi Ji. During my Ph.D. study, she has always been motivating, guiding, and encouraging me to be more fruitful and trained to be focused in more fundamental issues in my research. This thesis has significantly benefited from her guidance and encouragements. I can never thank her enough for her time and efforts devoted on my research and papers.

I would like to thank my thesis committee members, Professor Raghupathy Sivakumar, Professor John R. Barry, Professor Steven W. McLaughlin, and Professor Mustafa Ammar. Their remarks and comments have made significant contributions to improving the quality of my thesis. I also would like to thank Professor Vincent J. Mooney, serving as my proposal committee member. I am very thankful to the former and current members of the Communication Networks and Machine Learning (CNML) Laboratory, Guanglei Liu, Zesheng Chen, Noo, Rajesh, Jack for their helps, encouragements, and numerous discussions. They were great companions in my long journey to this thesis.

Finally but not the least, I would like to thank my parents and family for their endless supports. This thesis is dedicated to them.

# TABLE OF CONTENTS

# LIST OF TABLES

# LIST OF FIGURES

# SUMMARY

This work develops an analytical framework for distributed management of large networks where each node makes locally its management decisions. For example, in wireless networks, individual nodes locally adjust physical and logical configuration through information exchange with neighbors. Two issues remain open. One is the optimality, i.e., whether a distributed algorithm would result in a *near-optimal* network management. The other is the complexity, i.e., whether a distributed algorithm would scale gracefully with a network size. We study these issues through modeling, approximation, and randomized distributed algorithms.

In this thesis, we show *when* distributed management is nearly optimal. To do so, we first derive a global probabilistic model of a set of network management variables which characterizes the complex spatial dependence of the variables. The spatial dependence results from externally imposed management constraints and internal properties of communication environments (e.g., interference in the wireless channel). The global model is thus determined by these internal network characteristics and management requirements. We then apply probabilistic graphical models in machine learning to show when and whether the global model can be approximated by a local model. This study results in a sufficient condition for distributed management to be nearly optimal: the global model on a network management needs to be approximated within a given approximation error bound. We quantify the sufficient conditions for the near-optimality of the local model. We then show *how* to obtain a near-optimal configuration through decentralized adaptation of local configurations.

The sufficient conditions for the near-optimality of the local model depends on the communication networks and environments. We provide the sufficient near-optimality

conditions on both wireless and multi-class networks. We next derive a near-optimal distributed inference algorithm based on the local model.

For large wireless networks, the proposed approach is applied to (a) forming a 1-connected physical topology, (b) configuring a logical topology through spatially scheduling link activities, and (c) reconfiguring locally from failures. For multi-class networks, we characterize the trade-off between near-optimality and complexity of distributed decisions of policy-based resource allocation. In such networks, the routed paths of random source-destination pairs cause the spatial dependence on the decisions at nodes.

We validate our formulation and theory through simulations. We show that the distributed algorithm can obtain a near-optimal network management at a bounded communication cost; whereas, centralized schemes have a computation/communication cost that grows proportionally with the size of the network.

# CHAPTER I

# INTRODUCTION

## *1.1 Motivation*

As the network size increases, the management complexity gets increased accordingly. The management of large and complex networks is a challenging topic. For the feasibility of the management of large and complex networks, the management function should be scalable to the network size. The scalability can be achieved with distributed management. Distributed management can be defined as each node decides its own management parameters (e.g., transmission power, channel-access, position for wireless networks; rate-control, call-drop, call-preemption for wired core networks) independently and asynchronously based on the neighbors' information. Distributed network management is imperative for these large networks where each node needs to make decisions locally with information exchange only with neighboring nodes. For example, in wireless infrastructureless networks, each node adjusts locally its physical and logical configuration through information exchange with neighbors. Many heuristic algorithms have been developed with promising results in this area. However, two issues remain open. One is the optimality, i.e., whether a distributed algorithm would result in a near-optimal *network* management. The other is the complexity, i.e., whether a distributed algorithm would scale gracefully with a network size. We study these issues through modeling, approximation, and randomized distributed algorithms.

Modeling defines the optimality. We derive a global probabilistic model for a distributed network management which characterizes jointly the statistical spatial dependence of various network management variables of different layers. The model

is a Gibbs distribution that results from internal network properties (e.g., for wireless networks, node positions, wireless channels and interference); and external management constraints (e.g., for wireless networks, physical connectivity, signal quality and configuration costs). When the global model can be approximated within a given error bound by a local model, the local model is near-optimal. The complexity of the local model is characterized by the communication range among nodes. A trade-off between an approximation error and complexity results in sufficient conditions on near-optimality. When the probabilistic spatial dependence of links decays slowly, the scaling property is poor and results in an inefficient approximation by a local model. Otherwise, it results in a moderate scaling and an efficient approximation by a local model. For example, when the wireless channel decays slowly, the aggregated interference is large, and a node needs to communicate with $O(\sqrt[3]{N})$ neighbors for the resulting configuration to be near-optimal, where $N$ is the size of the network. This shows and an inefficient approximation by a local model. On the contrary, when the wireless channel decays rapidly, the aggregated interference is small, and a node only needs to communicate with $O(1)$ neighbors, resulting in an efficient approximation by a local model.

When the local model belongs to a Markov Random Field [33] [41], where the spatial dependence of network management variables shows the spatial Markov property, we can use a class of randomized distributed algorithms that are provided by Markov Random Fields. Due to complexity constraints, the allowable neighborhood size is usually bounded as a small value. Thus, we show the near-optimality conditions with a given neighborhood system as a function of complexity constraints, channel condition, node density, traffic distribution, and etc. If the dependency graph of the local model shows a spatial Markov dependence, distributed decisions based on the probabilistic network model asymptotically converge to the global optimal ones with moderate complexity [33]. These randomized distributed algorithms allow each node

to make distributed decisions based on information from the neighbors.

The distributed algorithms are applied to wireless infrastructureless networks with the examples of (a) forming a 1-connected physical topology, (b) configuring a logical topology that maximizes the spatial channel reuse, and (c) reconfiguring locally from failures. The algorithms are also applied to wired multi-class networks with the example of preemption-based resource allocations.

To understand the scalability of policy-based resource allocation in multi-class networks, we represent the resource allocation problem with a graph and convert the problem into a routing problem. The complexity of centralized and optimal decision is known to be NP-complete.

To cope with this complexity, the cause of complexity needs to be investigated. Moreover, to overcome this complexity, distributed approach is a necessity. We investigate whether and when the probabilistic model-based approach can achieve a near-optimal performance with a moderate complexity.

We validate our model, approximation and randomized distributed algorithms also through simulations. Hence, the objective of this research is to contribute to fundamental understanding of the near-optimality conditions for the scalable and distributed management of large networks.

## *1.2 Background and Related Work*

In this section, we review state-of-the-art techniques for distributed management of large wireless/multi-class networks.

### 1.2.1 Distributed Management and Optimality

The theoretical foundation of the distributed management can be found from the emergence theory [88]. It shows that a simple local interaction among neighboring

nodes can result in a complex group behavior of an autonomous system. The emergence property is found in many research areas: coherent pattern formation in physics and chemical systems [61], motion of swarm in biology [42], behavior of social groups, particular shape formation in control area [88], self-wiring of networks (e.g., neuron network, circuits), and autonomous self-configuration of wireless network topology. The emergence theory shows the possibility of distributed self-management of large autonomous systems.

The distributed network management provides a promising paradigm for the network self-management. In the ad-hoc wireless network, the main management parameters are transmission power, node position and mobility, and link activity. Some existing models and algorithms are proposed to manage these management parameters in a distributed fashion. The relative positions of mobile nodes are characterized with the swarm network models, and the self-configuration schemes are studied. The self-configuration scheme is derived by observing the local behavior of swarm groups: Levine *et. al* show the self-organization of self-propelled particles [61]; Junginger *et. al* study the self-organizing publish/subscribe middle-ware for dynamic peer-to-peer networks [54]; and Baras *et. al* show how to control autonomous swarms for certain formations [10]. These approaches are heuristic, and the performance measure compared to the optimal solution is not available. Mcdonald *et. al* provide a dynamic distributed clustering algorithm [72], which organizes the ad-hoc wireless network into clusters in which the probability of path availability is bounded. This work is not general but specifically focused on the studied problem.

Based on the wireless channel model, the distributed power control methods are proposed [30] [40]. Most work is done considering the fixed CDMA wireless network. Holliday *et. al* worked on self-optimizing the transmission power [40], and the transmission power of each node is optimized to satisfy the pre-defined signal-to-interference noise (SINR) ratio. Zuniga and Krishnamachari show the optimal

transmission power for minimizing the settling time in network flooding events. Chiang *et. al* [24] provide a distributed power control algorithm based on the sum product algorithm on graphs. For the multiple access control (MAC) layer, the link activities are decision variables. The contention graph and the conflict graph can describe the channel contention at the link layer [66], [62]. With conflict graph, Brar *et. al* [17] derive a centralized heuristic link-scheduling algorithm, which is based on the SINR physical channel contention model and whose complexity is only polynomial. [17] provides an approximation bound on its performance relative to optimal scheme and evaluates the proposed scheme extensively through simulation under representative wireless mesh network scenarios. However, as mentioned in [17], distributed scheme is still missing and provides an interesting possibility. Linear programming is used to find the optimal value of the link activities [83], which is a centralized scheme. Luo *et. al* [66] show the fair scheduling methods at the link layer of ad-hoc wireless networks. For the network and above layer, only little work has been done yet. The terminodes project [43] studies the self-organization of the ad-hoc wireless network, which means the network runs solely by the operation of end users. This project considers all layers and inter-layer interactions concurrently, and aims a paradigm shift in network management.

For the management of a large network with local cooperations in a random environment, the existing works may be insufficient. Some of existing work uses ad-hoc and heuristic methods based on empirical studies. They may work well sometimes; however, when and why they work is often not well-understood and needs to be characterized with the related parameters (e.g., channel condition $\alpha$, number of nodes $N$). Other existing works ask for satisfying too sufficient conditions, which costs redundant network resources.

To provide a solid understanding about the trade-off between optimality and complexity of distributed management, we use a bottoms-up approach that incorporates all related parameters in a single quantity.

### 1.2.2 Probabilistic Graphical Models: Spatial Dependence of Complex Systems

Graphical models have been used to represent the spatial dependence in complex systems. Hammersley-Clifford Theorem [33] shows that among various graphical models, an interesting type of probabilistic graphical models where a random variable is conditionally independent of the others given its neighbors is Markov Random Field. In particular, when the neighborhood is much smaller than the size of a network, the conditional independence implies an interesting type of spatial Markov dependence, i.e., a node depends on its far neighbors through neighbors' neighbors. Such a nested dependence can be shown explicitly through local connectivities among nodes in a dependency graph. The resulting probability distribution is thus factorizable in terms of local probability distributions.

In addition to their wide applications in image processing [33], Markov Random Fields have been used to model the cooperation of mobile agents [10]. One other related work is a cross-layer graphical model developed for optical networks [64]. There a Bayesian belief network models the attack propagation at the physical layer, and a Markov Random Field models the spatial dependence of routes at the network layer.

We propose to investigate the trade-off between near-optimality and complexity of the distributed management of large wireless (or wired multi-class) networks in a Markov Random Field.

### 1.2.3 Management of Wireless Network Configuration

The reliable communication of an active flow depends on both the physical and logical connectivity between the source and destination pair of the active flow [72]. The physical connectivity of an active flow indicates the existence of physical connectivity between intermediate nodes between the source and destination pair of the active flow. The logical connectivity means that the intermediate links satisfy the SINR requirement.

**Physical topology:** In the prior work, topology formation focuses on configuring a physical topology [69] and reconfiguring mobile sensor networks [102], and has been investigated for emerging behavior of mobile nodes [88] [10], intelligent agents [91], information management units, and mobile robots [21]. There, topology formation is usually treated through modeling the behavior of a system externally rather than internally. Hence the resulting model may not be able to characterize internal properties of a network.

**Logical topology:** In wireless LAN networks, every node resides in the transmission range of the other nodes, resulting in an efficient IEEE 802.11 RTS/CTS handshake [94] and a reliable link quality.

However, in wireless ad hoc networks, RTS/CTS handshake can fail to detect hidden nodes, which is due to heterogeneous transmission range among transmitters [97] [94]. Hidden and exposed terminal problems in ad hoc networks are more serious than in wireless LAN networks [94] [96]. [28] issues a potential problem of 802.11, which ignores interference outside a certain channel-contention range, which corresponds to the interference range in this work. This means that the logical link may not be reliable.

As these works investigate the interference due to the transmitters outside the

transmission range, little has been done to quantify jointly the effects of channel environments $\alpha$, densities of nodes and communication node-pairs, and optimality of distributed configuration management onto the reliability of logical links.

In this work, we show that both a physical and a logical topology are coupled and should be considered jointly. Such a joint treatment allows not only a physical topology but also a logical topology as well as their combinations to be configured in a fully distributed fashion. Another challenge is to provide a reliable transport network for active flows in a fully distributed fashion.

### 1.2.4 Management of Policy-based Resource Allocation

**Resource Allocation Policy:** Many resource allocation policies are proposed for the multi-class networks [11]. The guaranteed minimum policy makes each service class get its own small partition and the shared pool of resources; the upper limit policy puts an upper limit on the number of connections to each class; the complete partitioning policy allocates each class a set of resources that can only be used by the class; and the preemption policy allows a high-priority connection to preempt the active lower-priority connections.

Although many studies have been done about the resource allocation problem in the multi-class networks, the problem of strictly prioritized multi-class networks has not been studied enough (e.g., complexity of optimal decision). For the resource allocation problem in the strictly prioritized multi-class networks, the preemption policy is known to perform better than the representative upper limit policy [11][12].

**Preemption-based resource allocation:** When a network is so heavily loaded that the network cannot accept a new connection any more and the arrival patterns

of the connection request are unknown, or when a link or node failure happens, a connection preemption event may be triggered. The preempted flows may be rerouted, and the preempting and rerouting procedure continues until the lowest class is preempted and rerouted. The existing connection preemption algorithms decide which connections to preempt on the preempting route that is predetermined by a routing algorithm. The existing connection preemption algorithms are implemented either in a centralized or decentralized fashion. The complexity of preemption problem is known to be NP-complete [32], thus there are lots of practical and heuristic algorithms in the literature. However, the cause of complexity needs to be investigated, and distributed decisions can be applied to reduce complexity with a moderate performance degradation.

## *1.3 Problem Description*

In the wireless networks, as the network size increases, the management of the large wireless network is a challenging topic. For the feasibility of the management of large and complex networks, the management function should be scalable to the network size. The scalable wireless management can be achieved with *distributed management* (i.e., *independent and asynchronous decisions of individual nodes except local information exchange*). That is, distributed network management is imperative for these large networks where each node needs to make decisions locally with information exchange with neighboring nodes. For example, in wireless infrastructureless networks, each node adjusts locally its physical and logical configuration through information exchange with neighbors. However, as the network size increases, to maximize the resource utilization, the spatial reuse needs to be maximized. The spatial-reuse maximization increases the accumulated interference, which could make the distributed management fail to make the approximation error bounded. The distributed management in wireless networks can be characterized with the trade-off between the

near-optimality and complexity. These issues can be studied through modeling, approximation, and randomized distributed algorithms.

In the wired networks, the policy-based resource allocation in multi-class networks is known to have NP-complete complexity, thus most existing works are heuristic and decentralized ones. Centralized preemption is optimal but computationally intractable. Decentralized preemption is computationally efficient but may result in a poor performance. An open issue is to derive the complexity upper bound of the optimal preemption decision. Moreover, to understand the cause of complexity of the centralized and optimal preemption-based resource allocation in multi-class networks, we investigate the cause of complexity and investigate whether a distributed decision can achieve a near-optimality with a moderate complexity.

The other open issue is to study whether and when distributed preemption can achieve a near-optimal performance at a moderate complexity. The complexity originates from the spatial dependence of preemption decision variables at individual nodes. A challenge is how to characterize and coordinate efficiently a large number of spatially-dependent preemption-decision variables. Machine learning provides a framework to study these issues.

## *1.4 Our Approach*

In this thesis, to understand the scalable and near-optimal configuration management of large and complex wireless networks, we count on a model-based approach. Thus, our first step is to develop a probabilistic cross-layer network model which represents both the spatial dependence of the network state variables (e.g., node position, link channel-access, transmission power) and the affects of random environments (e.g., GPS position error, channel noise).

In our view, the optimality of configuration management should be considered

in a model-based framework. The reason is that if network configurations can be modeled, and a model characterizes the ground truth, optimality can be defined accordingly. Centralized configuration management can be used to define the optimality of a network configuration so that nodes are not limited by the scope of information exchange.

Three factors are considered in modeling of distributed management of a large network: randomness resulting from a network internally (e.g., wireless channel, interference), management constraints imposed externally (e.g., SINR, link-rate), and distributed decisions made by nodes (e.g., limited management information). For instance, randomness in wireless networks is challenging to model [52].

Nodes make asynchronous and randomized decisions in adjusting a configuration in a distributed setting. The model is developed from bottoms-up approach by mapping these three factors onto a probability distribution. The model is thus accurate in regard to the assumed communication environments, constraints and node decisions. Overall, such a model characterizes statistical spatial dependence of nodes/links in a network configuration.

To obtain an analytical form of the probabilistic model, we adopt an analogy between network decision variables (e.g., channel-access, flow-preemption-decision) of a network and interacting particles of a particle system in statistical physics [41][101]. The analogy allows us to apply a notion of "system Hamiltonian" [63] to quantify a network configuration. The Hamiltonian corresponds to an artificial system energy of a large network. The system energy combines both the internal randomness and external constraints into a single quantity. The Hamiltonian is then used to obtain a probabilistic model which is known as a Gibbs distribution [41]. Such a Gibbs distribution is for an entire "network" management decision (i.e., network configuration) and thus corresponds to a global probabilistic model.

The derived global model can be approximated with a local model. The near-optimality of distributed management depends on the near-optimality of the local model (i.e., whether the local model is accurate enough to the global model within an error bound). To investigate the near-optimality of the local model, we use the probabilistic graphical models and graph theory.

A Gibbs distribution can be represented with a probabilistic graphical model in machine learning [33][53]. The graph provides a simple and explicit representation of statistical spatial dependence in a network configuration. If the corresponding dependency graph of the local model exhibits a nested spatial Markov dependence, the approximation error corresponds to the ignored long range dependency links in the dependency graph.

How good is the approximation? The approximation error depends on communication environment (e.g., random routed paths of active flows, power decay of wireless channel, density of nodes, physical topology, and management constraints). For a large-size network, we investigate when and whether the approximation error resides within a desired error-bound as the network size increases. When the approximation error is within a given bound, a local model is near-optimal.

Once the local model belongs to a Markov Random Field, whose approximation error is within an error bound, we can use a class of randomized, distributed and iterative algorithms (e.g., stochastic relaxation, sum-product algorithm), where nodes self-configure through local information exchange with neighbors. The range of information exchange characterizes the local connectivity on the probabilistic dependency graph, and defines the communication complexity. Node decisions are probabilistic, corresponding to randomized distributed algorithms of graphical models. Distributed algorithms are applied into physical/logical topology formation and localized failure recovery.

To understand the feasibility of distributed resource allocation in multi-class networks, we propose to characterize the computation complexity of centralized and distributed decisions of resource allocation in multi-class networks. Especially, we study the preemption-based resource allocation in multi-class networks (such as multi-protocol label switching (MPLS) networks).

Connection preemption has been studied for the efficient resource management in the multi-class networks; however, there are only heuristic algorithms in the literature. We provide an upper-bound of the centralized optimal preemption problem, which is based on converting the preemption problem into a routing problem on a virtual graph. The complexity is known to be NP-complete.

To investigate the cause of complexity and the feasibility of near-optimality decisions, we count on a probabilistic distributed preemption decision, based on the machine learning approach.

## *1.5 Thesis Outline*

The thesis is organized as follows. In Chapter 2, we develop an analytical framework for the near-optimal distributed management of multi-hop wireless networks. We study the optimality of a network configuration through modeling. We begin with a global model that characterizes the ground truth in regard to assumptions of networks. The ground truth includes internal network characteristics on a wireless channel, a random physical and logical configuration, node decisions, and external management constraints. The model is a Gibbs distribution where the exponent can be regarded as a cost function. This relates the modeling with optimization. The global probabilistic model characterizes the spatial dependence of node positions and link activities with management constraints. The global model can be approximated with a local model.

Near-optimality conditions for such an approximation are derived under different channel conditions, density of nodes, network size and management requirements.

The optimality condition is coupled with the communication complexity which is the total number of interference neighbors in distributed self-configuration.

In Chapter 3, we derive a complexity upper-bound of the connection preemption problem. To do so, we propose to consider the preemption problem in the domain of a routing problem. Our approach is to represent the preemption problem with a flow graph (i.e., a *virtual network topology*), where each feasible route represents a feasible set of flows to be preempted. The least-cost route can be found with a least-cost routing algorithm; therefore, the complexity can be derived from that of the routing problem.

We also consider the preemption problems with both soft and strict (preemption) priority orders. With soft preemption order, the corresponding graph of the preemption problem is called the "multi-layer virtual topology" and we estimated its complexity as being very high. With strict priority orders, the preemption problem in a multi-class network can be segmented with multiple preemption problems of different priorities. The corresponding virtual topology of each priority is shown to be relatively simple. We compare these results with those obtained with distributed and centralized decisions.

In Chapter 4, we investigate distributed preemption where nodes make decisions whether and which flows to preempt using local information from neighboring nodes. Connection preemption is a key component for multi-class MPLS networks but known to be NP-complete. Centralized preemption is optimal but computationally intractable. Decentralized preemption is computationally efficient but may result in a poor performance. Our goal is to study whether and when distributed preemption can achieve a near-optimal performance at a moderate complexity. The complexity originates from the spatial dependence of preemption decision variables at individual

nodes. A challenge is how to characterize and coordinate efficiently a large number of spatially-dependent preemption-decision variables. Machine learning provides a framework to study these issues.

We first model a large number of distributed decisions using probabilistic graphical models in machine learning. We then define the near-optimality when the distributed preemption decisions approximate that of the optimal centralized preemption within a given error bound. We show that a sufficient condition for distributed preemption to be optimal is that local decisions constitute a Markov Random Field. The decision variables, however, do not possess an exact spatial Markov dependence in reality. Hence we derive sufficient conditions on traffic patterns of flows so that the distributed preemption is near-optimal at a cost of obtaining information from only a few neighbors.

We develop, based on the probabilistic graphical models, a near-optimal distributed algorithm. The algorithm is used by each node to make collective preemption decisions.

In Chapter 5, we conclude the thesis with discussions about the future direction of this research.

# CHAPTER II

# DISTRIBUTED CONFIGURATION MANAGEMENT OF WIRELESS NETWORKS

## *2.1 Introduction*

Wireless infrastructureless networks include sensor- and actor-networks, wireless mesh networks, and agent networks. A configuration of such a network is composed of a physical and a logical topology (configuration). A physical configuration is characterized by node positions and connectivity. A logical configuration is characterized by activities of links, i.e., a pattern of node-node communications on who is communicating with whom and when. For wireless networks, both a physical and a logical configuration can vary due to either failures or environmental changes. Configuration management is to adapt a physical and/or a logical topology to support and maintain active flows.

Generally, there is no centralized management authority for infrastructureless wireless networks. Hence self-configuration is desirable where nodes can adaptively adjust their own positions and communication patterns in a distributed fashion through local interactions. A challenge is whether distributed self-configuration would result in a near-optimal configuration with a sufficiently small approximation error.

In this work, we develop an analytical model for distributed self-configuration, and study the issue of near-optimality. For simplicity, we focus on ad-hoc wireless networks without mobility.

What is optimality? Deterministic optimization has been used to obtain an optimal solution for centralized management [15][24][28][68]. A key idea is to derive a

cost function that consists of management objectives and constraints. An optimal solution is obtained through global optimization of the cost function. Such an approach has been applied widely, e.g. to network capacity maximization through adapting a physical topology [37], and to link-scheduling through configuring a logical topology [15]. Such deterministic approaches, however, do not consider random factors such as inaccurate node positions, wireless channel, interference, and locally interacting wireless nodes. One other open issue is whether a cost function itself is optimal. This issue cannot be resolved in a conventional optimization framework.

From a computational standpoint, global optimization requires a centralized entity to maintain and update complete information for all nodes in the network. This is impractical for large networks. More importantly, locality can be generic to configuration management. For example, in a large wireless network, nodes and links often fail locally. A local repair is thus desirable for preventing an entire network from suffering incessant re-configurations. Therefore, distributed configuration management is a necessity for large wireless networks [15][52].

In a distributed setting, each node either adjusts its own physical position or decides when to transmit to whom based on local information. This should be done in a fully asynchronous and independent fashion with local information exchange. Local information on a configuration may include physical locations and communication activities (channel-access) of neighbors. Such information can be either sensed locally at a node or exchanged with neighbors. The range of information-exchange characterizes communication complexity. When the information-exchange is performed among only close neighbors, the resulting distributed algorithm would scale gracefully with a network size.

Numerous distributed algorithms and protocols have been developed for topology

formation [21][88][91] using local information (see [103] and references therein). Self-organizing protocols have been developed for sensor networks [29][79] and p2p self-stabilizing networks (see [58] and references therein). These distributed algorithms are deterministic. [10] is one of a few approaches that use a probabilistic model to characterize the randomness in node positions.

These distributed approaches provide promising empirical results. Yet the local rules used in the distributed algorithms are generally heuristic, and the performance of the algorithms is tested through simulation. Little has been done to quantify conditions on *when* and *how* the distributed management can result in a near-optimal configuration. It has been generally considered as a difficult problem to develop a distributed algorithm with a predictable performance [95]. Hence the open questions are

- *What* measures the optimality of centralized-configuration management, and the near-optimality of distributed-configuration management?
- *When* is it possible for distributed management to achieve a near-optimal configuration?
- *How* to derive distributed algorithms that obtain a near-optimal configuration?

This work intends to study these issues through *modeling*, *approximation*, and *algorithms*.

(a) **Global Model and Optimality**: In our view, the optimality of configuration management should be considered in a model-based framework. The reason is that if network configurations can be modeled, and a model characterizes the ground truth, optimality can be defined accordingly. Centralized configuration management can be used to define the optimality of a network configuration so that nodes are not limited by the scope of information exchange.

We consider three factors in a model of configuration, randomness resulting from a network internally, management constraints imposed externally, and distributed decisions made by nodes. Randomness in wireless networks is challenging to model [52]. This work begins with simple scenarios where the randomness results from random node positions, interference due to wireless channel conditions, and node-node communication. Fading is not considered in this work for simplicity. Management constraints include requirements on physical connectivity and signal quality, i.e., signal-to-interference plus noise ratio (SINR). Nodes make asynchronous and randomized decisions in adjusting a configuration in a distributed setting. The model is developed from bottoms-up by mapping these three factors onto a probability distribution. The model is thus accurate in regard to the assumed wireless channel, constraints and node decisions. Overall, such a model characterizes statistical spatial dependence of nodes/links in a network configuration.

To obtain an analytical form of the probabilistic model, we adopt an analogy between link activities and node positions of a wireless network and interacting particles of a particle system in statistical physics [41][101]. The analogy allows us to apply a notion of "configuration Hamiltonian" [63] to quantify a network configuration. The configuration Hamiltonian corresponds to an artificial system energy of a wireless network. The system energy combines the physical topology, link activities, and management constraints into a single quantity. The configuration Hamiltonian is then used to obtain a probabilistic model which is known as a Gibbs distribution [41]. Such a Gibbs distribution is for an entire "network" configuration and thus corresponds to a global probabilistic model.

(b) **Local Model and Probabilistic Graphs:** We then obtain an approximation of the global model referred to as a local model of network configurations. How to approximate the global model? There is a one-to-one mapping between a Gibbs

distribution and a probabilistic graphical model in machine learning [33][53]. The graph provides a simple and explicit representation of statistical spatial dependence in a network configuration.

We show that a probabilistic graph of the global model belongs to a two-layer random-field. One layer is for a random physical configuration, and the other is for a random logical configuration. The graph is fully connected, where the long-range spatial dependence results from the interference among far-away nodes due to wireless communication. When the long-range interference can be neglected, the global model can be approximated by a two-layer coupled Markov Random Field which is also called a Random Bond model. The corresponding dependency graph exhibits a nested spatial Markov dependence for both a physical and logical configuration. Mathematically, such a spatial Markov dependence can be represented as a product of local conditional probability density functions [33]. Hence the probabilistic graphical model shows which "dependency links" to remove, resulting in a local model.

How good is the approximation? We define an approximation error between the local and global model. The approximation error depends on power decay of wireless channel, density of nodes, physical topology, and management constraints. For a large-size network, we show that the approximation error resides within a desired error-bound when the total interference from far-away nodes decreases faster than the growth of the number of interfering nodes as the network size increases. When the approximation error is within a given bound, a local model is near-optimal.

(c) **Distributed Algorithm**: A local model, i.e., Markov Random Field, allows a class of distributed algorithms where nodes self-configure through local information exchange with neighbors. The range of information exchange characterizes the local connectivity on the probabilistic dependency graph, and defines the communication complexity. The actual information exchanged includes relative positions of neighbors

and activities of adjacent nodes. Node decisions are probabilistic, corresponding to randomized distributed algorithms of graphical models.

We apply the distributed algorithm to examples in three aspects of self-configuration: (a) forming a 1-connected physical topology from a random initial topology; (b) configuring a logical topology that maximizes spatial channel-reuse and incessant communication demands; and (c) reconfiguring a physical- and logical-configuration upon failures. Configuring a logical topology is done in the context of scheduling [31]. This framework would allow fully distributed spatial scheduling algorithms and fault tolerance for wireless infrastructureless networks.

## *2.2 Problem Formulation*

### 2.2.1 Assumptions

Consider a wireless network with the following assumptions.

*Physical Layer:* All nodes share a common frequency channel[1]. A pair of nodes within a communication range can communicate directly with an omni-directional antenna. The wireless channel follows a path-loss model with a power attenuation of factor $\alpha \in \{2 \sim 6\}$. For simplicity, shadowing and/or multi-path fading are not considered in this work. Node $i$ transmits with power $P_i$, where $0 \leq P_i \leq P_{max}$, $1 \leq i \leq N$, with $P_{max}$ being the maximum transmission power, and $N$ being the number of nodes in the network. Power control is not considered in this work.

*MAC Layer:* Let $\text{SINR}_{th}$ be a given threshold for the SINR requirement. Node $i$ can transmit to node $j$ when the SINR requirement is satisfied, i.e., $\text{SINR}_{ij} =$

[1]An extension to multiple channels is straightforward, and will be considered in a subsequent work.

$\frac{P_i l_{ij}^{-\alpha}}{N_b + \sum_{(m,n)\neq(i,j)} P_m l_{mj}^{-\alpha}} \geq \text{SINR}_{th}$, where $l_{ij}$ is the distance between node $i$ and $j$, and $N_b$ is noise power. We consider a scheduled resource allocation that is implemented with local interactions among neighbors.

*Configuration Management:* Consider a wireless network with $N$ nodes. Let $X_{i0}$ and $X_i$ be a desired and an actual location of node $i$, for $1 \leq i \leq N$. $\boldsymbol{X} = \{X_1, \cdots, X_N\}$ are random positions of nodes in a network where the randomness results from measurement errors, perturbed positions, and random movements.

Let $\sigma_{ij}$ denote channel-access of link $(i, j)$, where $\sigma_{ij} = 1$ if node $i$ is transmitting to node $j$; and $\sigma_{ij} = -1$, otherwise. $\sigma_{ij}$ is referred to as a "communication dipole" in this work, for $1 \leq i, j \leq N$, $i \neq j$, and $\boldsymbol{\sigma} = \{\sigma_{1,2}, \cdots, \sigma_{N,N-1}\}$ denotes a set of link activities in the network. Link activities are assumed to be random as they are triggered by network-layer random traffic demands. A logical configuration is $\boldsymbol{\sigma} = \{\sigma_{ij}\}$. A network configuration is $(\boldsymbol{\sigma}, \boldsymbol{X})$.

The objectives are to achieve distributed configuration management, i.e., to (a) form a desired physical topology, (b) schedule the resource utilization at a given time to maximize the spatial channel-reuse with a desired SINR requirement, and (c) reconfigure upon failures by minimizing a reconfiguration cost.

### 2.2.2 Formulation

Let $\text{P}(\boldsymbol{\sigma}, \boldsymbol{X})$ be a true probabilistic model of a network configuration that results from the above assumptions. The model is referred to as the global model.

Definition 1. *Optimal Configuration: $(\boldsymbol{\sigma}^*, \boldsymbol{X}^*)$ is an optimal configuration if it maximizes the global likelihood,*

$$(\boldsymbol{\sigma}^*, \boldsymbol{X}^*) = arg \max_{(\boldsymbol{\sigma}, \boldsymbol{X})} P(\boldsymbol{\sigma}, \boldsymbol{X}). \quad (1)$$

Let $P^l(\boldsymbol{\sigma}, \boldsymbol{X})$ be an approximation of the probabilistic network model $\mathrm{P}(\boldsymbol{\sigma}, \boldsymbol{X})$. $P^l(\boldsymbol{\sigma}, \boldsymbol{X})$ is referred to as the local model.

Definition 2. *Near-Optimal Configuration: Consider* $(\hat{\boldsymbol{\sigma}}, \hat{\boldsymbol{X}})$ *that maximizes* $P^l(\boldsymbol{\sigma}, \boldsymbol{X})$, *i.e.,*

$$(\hat{\boldsymbol{\sigma}}, \hat{\boldsymbol{X}}) = arg \max_{(\boldsymbol{\sigma}, \boldsymbol{X})} P^l(\boldsymbol{\sigma}, \boldsymbol{X}). \tag{2}$$

*Consider an approximation error as the average relative difference between the log likelihoods,*

$$E[\Delta] = E\left[\left|\frac{\log P(\boldsymbol{\sigma}^*, \boldsymbol{X}^*) - \log P(\hat{\boldsymbol{\sigma}}, \hat{\boldsymbol{X}})}{\log P(\boldsymbol{\sigma}^*, \boldsymbol{X}^*)}\right|\right]. \tag{3}$$

*For a given* $\epsilon > 0$, *if* $E[\Delta] \leq \epsilon$, $(\hat{\boldsymbol{\sigma}}, \hat{\boldsymbol{X}})$ *is near-optimal.*

Distributed configuration management requires that $P^l(\boldsymbol{\sigma}, \boldsymbol{X})$ is factorizable, i.e., $P^l(\boldsymbol{\sigma}, \boldsymbol{X})$=$\prod_{ij}\ g_{ij}(\boldsymbol{\sigma}, \boldsymbol{X})$, where $g_{ij}(\boldsymbol{\sigma}, \boldsymbol{X})$ is a localized probability density function that depends on variables in a neighborhood of node $i$, node $j$ and link $(i, j)$ for $1 \leq i, j \leq N$. The global maximization from Eq.(1) would reduce to a set of coupled local maximizations, i.e., $(\hat{\sigma}_{ij}, \hat{X}_i) = \arg\max_{(\sigma_{ij}, X_i)}\ g_{ij}(\boldsymbol{\sigma}, \boldsymbol{X})$ for $1 \leq i, j \leq N$.

Our tasks are to

(a) obtain a global model $\mathrm{P}(\boldsymbol{\sigma}, \boldsymbol{X})$ from the above given assumptions;

(b) represent the spatial dependence of a configuration using a probabilistic graphical model for $\mathrm{P}(\boldsymbol{\sigma}, \boldsymbol{X})$. Obtain a simplified graph and a mathematical representation for $P^l(\boldsymbol{\sigma}$,$\boldsymbol{X})$;

(c) obtain sufficient conditions for $P^l(\boldsymbol{\sigma}$,$\boldsymbol{X})$ to result in a near-optimal configuration;

(d) derive a distribution algorithm, and apply the algorithm to examples of self-configuration.

## *2.3 Global Model*

We begin by developing a global model that characterizes probabilistic spatial dependence in a network configuration. Our approach is bottoms-up so that the probabilistic model can be obtained faithfully based on given assumptions and management constraints.

### 2.3.1 Logical Configuration

We begin with modeling a logical configuration given node positions.

#### *2.3.1.1 Example*

First consider through a simple example why a logical configuration should be considered as random. Consider a linear network in Figure 1. Assume that the channel-contention requires any two active links to be separated by at least two silent links. Assume that the network achieves the spatial channel-reuse maximization, i.e., the total number of concurrent active links is maximized through either centralized or distributed algorithms. Centralized decision is done by a node with the complete information on a physical topology and link activities. In contrast, distributed decisions are done at each node iteratively with only local information exchange with neighbors.

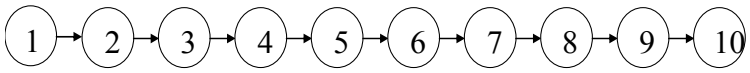



**Figure 1:** An Example Network with 10 Nodes and 9 Directional Links.

A configuration of link activities is triggered by network-layer traffic demands (random flows) [64]. Figure 2 shows all possible logical configurations given a physical topology and all-to-all traffic demands. Each row corresponds to a snapshot of active links at a time epoch, and those active links should satisfy the aforementioned

contention requirement[2]. For example, the first row denotes a configuration where both link $(3, 4)$ and $(8, 9)$ are active. There are 2 configurations with two active links, and 8 configurations with three active links.

For centralized decisions where each node knows the activities of the others, only the configurations with three active links are feasible. If all these configurations were regarded equally likely, there would be multiple configurations that maximize the spatial reuse equally well. Hence which configuration to choose would not be unique but random.

For distributed decisions where each node only knows the activities of its neighbors, the patterns for two active links are also feasible. Which of the two patterns appears depends on which node decides to transmit first, which can be random. Hence, logical configurations are random in general for both centralized and distributed decisions.

An additional issue is how to find such an optimal configuration through fully distributed node decisions with only local information. For example, if two links $(2, 3)$ and $(7, 8)$ happen to be active, is it possible to result in a pattern of 3 active links through distributed decisions? The answer is *yes* if the distributed algorithms are randomized, i.e., nodes make probabilistic decisions for transmission. We shall soon explain this in Section VI.

#### *2.3.1.2 Configuration Hamiltonian*

We now consider a general wireless network and develop a probabilistic model for logical configuration $\boldsymbol{\sigma}$ assuming a given set of node positions $\boldsymbol{X}$. $\boldsymbol{\sigma}$ is regarded as a set of communication dipoles. Terminology "dipole" is originally used in a particle system in statistical physics. There, a dipole corresponds to a particle with binary

[2]The contention requirement does not need to be satisfied at different time epochs, i.e., across different rows.

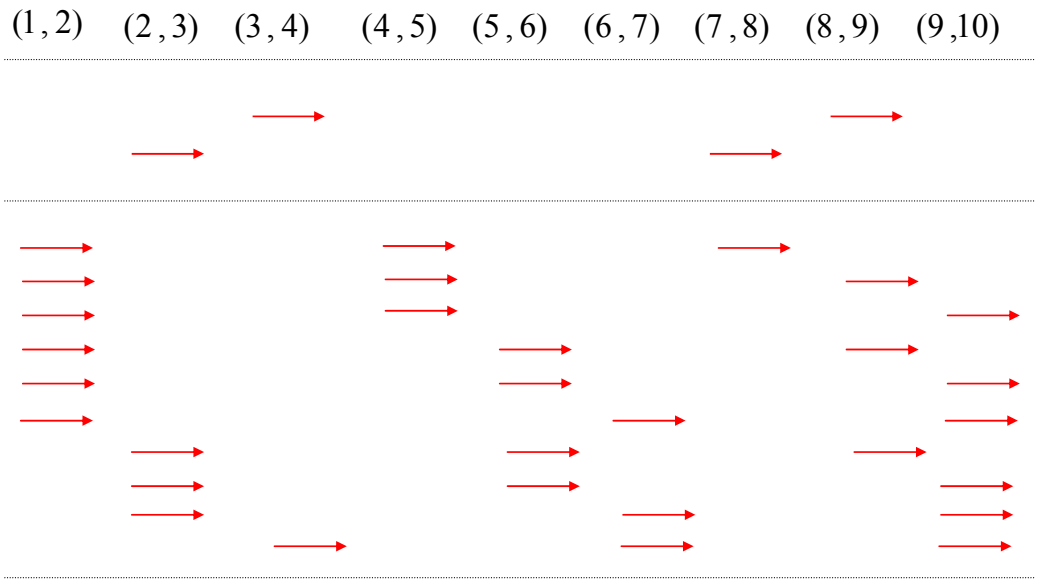



**Figure 2:** Possible configurations. "→": Active link

states, active or inactive [63]. Now consider each "communication dipole" as a particle. Table *I* compares a wireless network with a particle system, demonstrating their similarities.

**Table 1:** Correspondence between Dipole System and Lattice Gas

| Dipole System | Lattice Gas [63] |
|---|---|
| active(+1) / inactive(-1) | occupied(+1) / empty(-1) |
| interference | interaction energy |
| system potential energy | chemical potential |
| logical configuration | system state (e.g., either liquid or gas) |

Configuration Hamiltonian has been applied to a particle system to describe the states of a set of spins under the following conditions: (a) active particles are statistically distinguishable, and (b) interactions between particles are weak.

We now extend the notion of configuration Hamiltonian to the wireless network, where active communication dipoles are statistically distinguishable; and interactions

among dipoles are weak due to decaying interference. We define a system energy of a logical configuration as the summation of the received communication power at individual receivers in the network, i.e.,

$$\sum_{ij} P_j \cdot \frac{\sigma_{ij}+1}{2}, \tag{4}$$

where $P_j$ denotes the net received-power at the receiver $j$ by considering all interferences and noise. Based on the assumptions given in Section 2.2, for a single active dipole $\sigma_{ij} = 1$ in the network, the received power at the receiver $P_j = P_i l_{ij}^{-\alpha} \ \frac{\sigma_{ij}+1}{2}$, where $P_i$ is the transmission power at transmitter $i$ and $l_{ij} = |X_i - X_j|$. A dipole is inactive, i.e., $\sigma_{ij} = -1$ and $P_j = 0$, if node $i$ does not transmit to node $j$.

Following the definitions in statistical physics [41], the "configuration Hamiltonian" of a dipole system is the negative system energy [51],

$$H(\boldsymbol{\sigma}|\boldsymbol{X}) = -\sum_{ij} P_j \eta_{ij} + \beta \sum_{ij} (\mathrm{SINR}_{ij} - \mathrm{SINR}_{th})^2 \eta_{ij}, \tag{5}$$

where $\eta_{ij} = \frac{\sigma_{ij}+1}{2}$, $\mathrm{SINR}_{ij} = \frac{P_i l_{ij}^{-\alpha} \eta_{ij}}{\sum_{mn \neq ij} P_m l_{mj}^{-\alpha} \eta_{mn} + N_b}$ is the SINR for dipole $\sigma_{ij}$, and $\beta$ is a positive constant. $\beta(\mathrm{SINR}_{ij} - \mathrm{SINR}_{th})^2$ serves as a penalty term for the SINR constraint[3], an equivalence of which is $\beta[P_i l_{ij}^{-\alpha} \eta_{ij} - \mathrm{SINR}_{th}(\sum_{mn \neq ij} P_m l_{mj}^{-\alpha} \eta_{mn} + N_{b_{ij}})]^2$.

For an active dipole $\sigma_{ij} = 1$, the interference sources within a certain neighborhood from the receiver $j$ are considered as the significant interferers; and this neighborhood is denoted by $N_{ij}^I$ as the interference range of node $j$. The Hamiltonian can be rewritten as

$$H(\boldsymbol{\sigma}|\boldsymbol{X}) = R_1(\boldsymbol{\sigma}, \boldsymbol{X}) + R_2(\boldsymbol{\sigma}, \boldsymbol{X}) + R_3(\boldsymbol{\sigma}, \boldsymbol{X}) + R_I(\boldsymbol{\sigma}, \boldsymbol{X}), \tag{6}$$

where $R_1(\boldsymbol{\sigma}, \boldsymbol{X}) = \sum_{ij} \alpha_{ij} \eta_{ij}$ is the first-order energy of individual dipoles, $R_2(\boldsymbol{\sigma}, \boldsymbol{X})$ $= \sum_{ij} \sum_{mn \in N_{ij}^I} \alpha_{ij,mn} \eta_{ij} \eta_{mn}$ is the second-order energy with products of two dipoles

[3]In the analysis and distributed algorithms, we relax this strict constraint with $\beta U(\mathrm{SINR}_{th} - \mathrm{SINR}_{ij})$ where $U(x) = 1$ for $x > 0$; 0, otherwise.

within the interference range, $R_3(\boldsymbol{\sigma}, \boldsymbol{X}) = \sum_{ij} \sum_{mn \in N_{ij}^I} \sum_{uv \in \{N_{ij}^I, N_{mn}^I\}} \alpha_{ij,mn,uv} \eta_{ij} \eta_{mn} \eta_{uv}$ is the third-order energy with products of three dipoles within the interference range, $R_I(\boldsymbol{\sigma}, \boldsymbol{X}) = \sum_{ij} R_{I_{ij}}(\boldsymbol{\sigma}, \boldsymbol{X})$ is the total interference outside the interference range where $R_{I_{ij}}(\boldsymbol{\sigma}, \boldsymbol{X})$ is the residual interference outside the interference range of an active dipole $\sigma_{ij} = 1$.

The coefficients for the link activities $\boldsymbol{\sigma}$ depend on relative node positions $l_{ij}$'s, where

$$\begin{aligned} \alpha_{ij} &= -P_i l_{ij}^{-\alpha} + \beta \cdot (P_i l_{ij}^{-\alpha} - \mathrm{SINR}_{th} N_b)^2, \qquad (7) \\ \alpha_{ij,mn} &= 2\sqrt{P_i P_m} l_{ij}^{-\frac{\alpha}{2}} l_{mj}^{-\frac{\alpha}{2}} - P_m l_{mj}^{-\alpha} + \beta \mathrm{SINR}_{th}^2 P_m^2 l_{mj}^{-2\alpha} \\ &\quad -2\beta(P_i l_{ij}^{-\alpha} - \mathrm{SINR}_{th} N_b) \cdot \mathrm{SINR}_{th} P_m l_{mj}^{-\alpha}, \\ \alpha_{ij,mn,uv} &= -2\sqrt{P_m P_u} l_{mj}^{-\frac{\alpha}{2}} l_{uj}^{-\frac{\alpha}{2}} + \beta(\mathrm{SINR}_{th}^2 P_m P_u l_{mj}^{-\alpha} l_{uj}^{-\alpha}). \end{aligned}$$

Intuitively, $\alpha_{ij}$ corresponds to the increased power when dipole $\sigma_{ij}$ becomes active, $\alpha_{ij,mn}$ relates to the interference experienced by $\sigma_{ij}$ resulting from a neighboring active dipole $\sigma_{mn}$, and $\alpha_{ij,mn,uv}$ relates to the interference experienced by $\sigma_{ij}$ from both $\sigma_{mn}$ and $\sigma_{uv}$.

### 2.3.2 Physical Configuration

Now consider node positions $\boldsymbol{X}$. $\boldsymbol{X}$ is random where the randomness originates from either perturbed or estimated node locations with measurement errors. One example model for the noise/perturbation is a Gaussian distribution. For example, node positions can be characterized by a multi-variant Gaussian distribution $P(\boldsymbol{X})$ with an exponent $\sum_i \frac{(X_i - X_{i0})^2}{2\sigma^2}$, where $\sigma^2$ is the variance, and $\boldsymbol{X_0} = \{X_{i0}\}$ for $1 \leq i \leq N$ is a set of desired positions.

Management constraints are imposed on the physical connectivity. The 1-connectivity is an example of a constraint to achieve the reachability of any source-destination pairs in the network. That is, there exists at least one connected path between any two nodes in the network. A sufficient condition of the 1-connected physical topology is to construct a physical topology similar to the Yao graph, where each node has a connected link with its nearest neighbors every $\theta$ ($\leq \frac{2\pi}{3}$) radian apart [103]. Such a constraint can be represented as

$$h(X_i, X_j) = \begin{cases} 0, & |\frac{l_{ij}-l_{th}}{l_{th}}| \leq \epsilon_0, \\ |l_{ij} - L_{ij}|, & \text{otherwise}, \end{cases} \tag{8}$$

where $\epsilon_0$ is a small positive constant, $L_{ij}$=$|X_{i0} - X_{j0}|$ is the desired distance [4] of $l_{ij}$, $\forall\ (i, j)$. The resulting Hamiltonian for the physical topology is

$$H(\boldsymbol{X}) \ = \ \sum_i \frac{(X_i - X_{i0})^2}{2\sigma^2} + \zeta \sum_i \sum_{j \in N_i^\theta} h(X_i, X_j), \tag{9}$$

where $N_i^\theta$ is the set of the nearest neighbors of node $i$ for every $\theta$ radian, and $\zeta$ is a positive weighting constant.

### 2.3.3 Network Configuration

We now consider a network configuration which consists of both a physical and a logical configuration.

#### *2.3.3.1 Network Configuration Hamiltonian*

Combining the Hamiltonians for the physical and logical configurations results in an overall configuration Hamiltonian of a wireless network,

$$H(\boldsymbol{\sigma}, \boldsymbol{X}) = \varsigma_\sigma H(\boldsymbol{\sigma}|\boldsymbol{X}) + (1 - \varsigma_\sigma) H(\boldsymbol{X}), \tag{10}$$

where $\varsigma_\sigma$ is a scaling constant, $0 \leq \varsigma_\sigma \leq 1$, for the relative importance of these two Hamiltonians [63][10].

[4] for example, $L_{ij} = l_{th}$ for a positive constant $l_{th} > 0$.

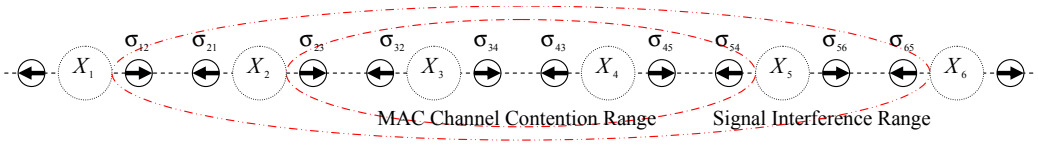



**Figure 3:** Interference Neighborhood on A Linear Network

#### *2.3.3.2 Gibbs Distribution*

A configuration Hamiltonian can then be related to a probabilistic model through a Gibbs distribution [63]. To be specific, in a particle system [41], the effective system potential energy $H(\omega)$, known as the configuration Hamiltonian, obeys the Gibbs (or Boltzmann) distribution [41], $P(\omega) = Z_0^{-1} \cdot \exp\left(\frac{-H(\omega)}{T}\right)$, where $Z_0$ is a normalizing constant and $T$ is the temperature of the particle system [33][41].

**MAC-layer model:** For a logical configuration $\boldsymbol{\sigma}$ given node positions $\boldsymbol{X}$, a Gibbs distribution $P(\boldsymbol{\sigma}|\boldsymbol{X})$ can be obtained using configuration Hamiltonian $H(\boldsymbol{\sigma}|\boldsymbol{X})$,

$$P(\boldsymbol{\sigma}|\boldsymbol{X}) = Z_\sigma^{-1} \cdot \exp\left(\frac{-H(\boldsymbol{\sigma}|\boldsymbol{X})}{T}\right), \tag{11}$$

where $Z_\sigma = \sum_{\boldsymbol{\sigma}} \exp\left(\frac{-H(\boldsymbol{\sigma}|\boldsymbol{X})}{T}\right)$ is a normalizing constant and also called the partition function [33]. $T > 0$ is the temperature in statistical physics that characterizes the stability of the system[5]. The lower the temperature, the more stable the configuration is. $T$ is used in [33] as a computational variable to obtain a most probable configuration. We shall follow this application in Section 2.6.

**Physical-layer model:** Similarly, the Gibbs distribution of node positions can be obtained as

$$P(\boldsymbol{X}) = {Z_X}^{-1} \cdot \exp\left(\frac{-H(\boldsymbol{X})}{T}\right), \tag{12}$$

[5]generally used in simulated annealing [10][33]

where $Z_X$=$\sum_{\boldsymbol{X}} \exp\left(\frac{-H(\boldsymbol{X})}{T}\right)$ is a normalizing constant.

**Cross-layer network model:** The Gibbs distribution of an entire network configuration can be obtained using the overall configuration Hamiltonian

$$P(\boldsymbol{\sigma}, \boldsymbol{X}) = {Z_0}^{-1} \cdot \exp\left(\frac{-H(\boldsymbol{\sigma}, \boldsymbol{X})}{T}\right), \tag{13}$$

where $Z_0$=$\sum_{(\boldsymbol{\sigma},\boldsymbol{X})} \exp\left(\frac{-H(\boldsymbol{\sigma},\boldsymbol{X})}{T}\right)$ is a normalizing constant.

#### *2.3.3.3 Minimum Hamiltonian and Optimal Configuration*

An optimal configuration is the one that maximizes the likelihood function,

$$(\boldsymbol{\sigma}^*, \boldsymbol{X}^*) = \arg\max_{(\boldsymbol{\sigma},\boldsymbol{X})} P(\boldsymbol{\sigma}, \boldsymbol{X}), \tag{14}$$

where $H(\boldsymbol{\sigma}, \boldsymbol{X}) = -\log\left[P(\boldsymbol{\sigma}, \boldsymbol{X})\right]/T \; -\log\left(Z_0\right)$. Note that $Z_0$ is a normalization constant.

Note the system energy $H(\boldsymbol{\sigma}, \boldsymbol{X})$ incorporates both the randomness and the external management requirements. When the constraints are satisfied, the penalty terms should be diminishing. Therefore, an optimal configuration should satisfy the management objectives, i.e., spatial reuse, and the constraints.

## *2.4 Local Model: Cross-Layer Markov Random Field*

We now seek a local model $P^l(\boldsymbol{\sigma}, \boldsymbol{X})$ that is factorizable by a product of localized probability density functions, and is a good approximation to the global model $P(\boldsymbol{\sigma}, \boldsymbol{X})$. We resort to probabilistic graphical models for this examination.

### 2.4.1 Graphical Representation

Probabilistic graphical models relate a probability distribution with a dependency graph of the corresponding random variables [33][53][55]. A node in the graph represents a random variable and a link between two nodes characterizes their statistical

dependence. In particular, a set of random variables $\boldsymbol{v}$ forms Gibbs Random Field (GRF) if it obeys a Gibbs distribution [63]. Hammersley-Clifford theorem shows an equivalence between a probabilistic dependency graph and a Gibbs distribution; and more importantly, when a Gibbs distribution has spatial Markov properties.

Hammersley-Clifford Theorem [63]: *Let $S$ be the set of nodes $S = \{1, \cdots, N\}$. Let $\boldsymbol{v}$ be a set of random variables $\boldsymbol{v} = \{v_1, \cdots, v_N\}$. $\boldsymbol{v}$ is said to be a Markov Random Field if (i) $P(\boldsymbol{v}) > 0$ for $\forall$ $\boldsymbol{v}$ in sample space; (ii) $P(v_i|\boldsymbol{v_j}$ for $j \in S\backslash\{i\}) = P(v_i|\boldsymbol{v_j}$ for $j \in N_i)$. $N_i$ is a set of neighboring nodes of node $i$ for $1 \leq i \leq N$.*

*Random field $\boldsymbol{v}$ is also said to be a Gibbs Random Field if its probability distribution can be written in the form $P(\boldsymbol{v}) = \prod_{c \in C} V_c(\boldsymbol{v})$, where $c$ is a clique, $C$ is the set of all feasible cliques, and $V_c(\boldsymbol{v})$ is a general positive function.*

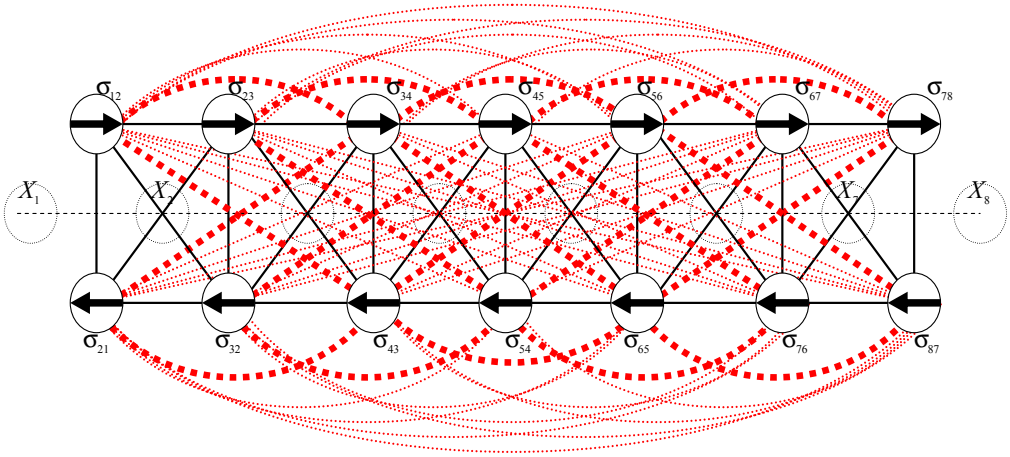



**Figure 4:** Spatial Dependency Graph of Link Activities $\boldsymbol{\sigma}$ given a Set of Node Positions $\boldsymbol{X}$

Hammersley-Clifford Theorem asserts that $\boldsymbol{v}$ is a Markov Random Field *if and only if* the probability distribution P($\boldsymbol{v}$) follows a Gibbs distribution. This theorem shows an interesting type of probabilistic graphical models where a random variable is conditionally independent of the others given its neighbors. In particular, when the

neighborhood is much smaller than the size of a network, the conditional independence implies an interesting type of spatial Markov dependence, i.e., a node depends on its far neighbors through neighbors' neighbors. Such a nested dependence can be shown explicitly through local connectivities among nodes in a dependency graph. The resulting probability distribution is thus factorizable in terms of local probability distributions.

For illustration, consider a one-dimensional topology as in Figure 3. The corresponding dependency graph for link activities $\boldsymbol{\sigma}$ given a set of node positions $\boldsymbol{X}$ is shown in Figure 4. Nodes in the graph represent binary random variables $\sigma_{ij}$'s, and the links represent their spatial dependence. For example, solid lines show the dependence due to channel contention, and the dashed lines indicate the dependence due to interference. The first and second rows of dipoles show bidirectional link-activities.

All communication dipoles are fully connected due to interference, and thus spatially dependent. This spatial dependence can be represented with the Hamiltonian in Eq.(6) that consists of the products of all pairs of dipoles. This fully connected dependency graph shows an uninteresting case of random field where the neighborhood of a node on the graph is the entire network. This implies that obtaining an optimal configuration may require each node to exchange information with all the other nodes in the network.

### 2.4.2 Approximation

How to obtain a meaningful approximation to the global model? The interference outside the interference range, $R_I(\boldsymbol{\sigma}, \boldsymbol{X})$, can be relatively small compared to the first three terms of the configuration Hamiltonian in Eq.(6). The third-order term, $R_3(\boldsymbol{\sigma}, \boldsymbol{X})$, can be small also compared to the second-order term. Hence, if we use the

first two terms to approximate the configuration Hamiltonian, we have

$$H^l(\boldsymbol{\sigma}|\boldsymbol{X}) = \sum_{ij} \alpha_{_{ij}}(\boldsymbol{X})\eta_{ij} + \sum_{ij} \sum_{mn \in N_{ij}^I} \alpha_{_{ij,mn}}(\boldsymbol{X})\eta_{ij}\eta_{mn}, \tag{15}$$

and the corresponding Gibbs distribution is

$$P^l(\boldsymbol{\sigma}|\boldsymbol{X}) = Z_l^{-1} \cdot \exp\left(\frac{-H^l(\boldsymbol{\sigma}|\boldsymbol{X})}{T}\right), \tag{16}$$

where $Z_l$ is a normalization constant.

As the sum in Eq.(15) only involves neighboring dipoles which are within the interference range, the resulting dependency graph now has a small neighborhood (see the thick dashed-lines in Figure 4). In fact this approximated Markov Random Field is the well-known second-order Ising model [63] where the Hamiltonian $H^l(\boldsymbol{\sigma}|\boldsymbol{X})$ consists of both the first- and second-order terms of $\sigma_{ij}$'s. Such an approximation can also be obtained directly from the probabilistic dependency graph of the global model. That is, by removing all edges outside the interference range for each node, we can obtain the graphical representation of the local model.

### 2.4.3 Spatial Dependence in Physical Topology

We now examine the spatial dependence of node positions $\boldsymbol{X}$. In general, the random field of node positions $\boldsymbol{X}$ may not be Markov and thus corresponds to a fully connected graph. However, key management objectives often result in physical topologies with spatial Markov dependence. For example, under the 1-connectivity constraint such as in Eq.(8), node positions $\boldsymbol{X}$ correspond to a second-order Markov Random Field, where the interactions are only with the first-order neighbors [6] that are $\theta$ radian apart.

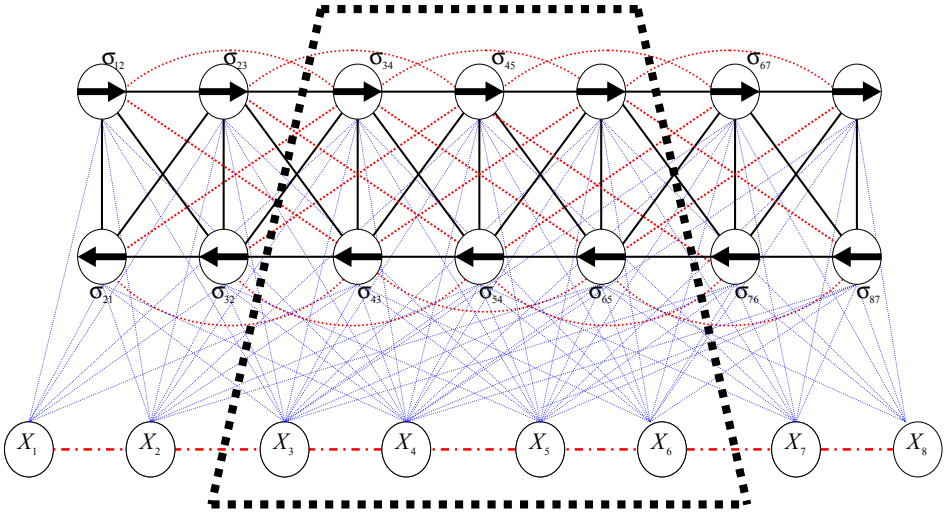



**Figure 5:** Cross-Layer Coordination Graph and Clique of ($\boldsymbol{\sigma}$, $\boldsymbol{X}$). The Box: A Clique of the Cross-Layer Graph

### 2.4.4 Cross-layer Markov Random Fields

The probabilistic graph of an overall configuration can be obtained by combining two graphs for the logical and physical configuration. For the example network, a cross-layer graph is shown in Figure 5 as an approximation of the original overall configuration, where for the upper layer graph of the logical configuration, the local interference is assumed among neighboring active dipoles. The lower-layer graph, which is for the physical configuration, shows Markov dependence of node positions due to the 1-connectivity constraint in Eq.(8). The entire graph is thus locally-connected with the spatial Markov dependence at both layers.

This cross-layer graph corresponds to a coupled MRF [33], where both an Ising model and a second-order MRF are combined together. The graph is also known as a Random-Bond model [63], where dipoles are connected by random bonds which depend on node positions. The cross-layer MRF ($\boldsymbol{\sigma}$, $\boldsymbol{X}$) can also be represented by a

[6]In this work, we assume those topology constraints result in spatial Markov node positions in order to focus on the impact of interference. Non-Markov constraints will be studied in future work.

chain graph [64] of two MRF blocks, one for $\boldsymbol{X}$ and the other for $\boldsymbol{\sigma}$. The cross-layer probabilistic graph thus maps the complex spatial dependence of a multi-hop wireless network to an explicit graphical representation.

The corresponding likelihood function can then be represented as

$$P^l(\boldsymbol{\sigma}, \boldsymbol{X}) \propto \prod_{i,j} g_{ij}(\boldsymbol{\sigma}, \boldsymbol{X}), \tag{17}$$

where $g_{ij}(\boldsymbol{\sigma}, \boldsymbol{X})$ is a local probability density function and can be represented as a function of the sum of clique potentials:

$$\begin{aligned} g_{ij}(\boldsymbol{\sigma}, \boldsymbol{X}) &= \prod_{c \in C_{ij}} \exp\left(\frac{-\psi_c(\boldsymbol{\sigma}, \boldsymbol{X})}{T}\right) \\ &= \exp\left(\frac{-\sum_{c \in C_{ij}} \psi_c(\boldsymbol{\sigma}, \boldsymbol{X})}{T}\right), \end{aligned}$$

where $C_{ij}$ is the set of all cliques including node $i$, node $j$, and link $(i, j)$; and $\psi_c(\boldsymbol{\sigma}, \boldsymbol{X})$ is a clique potential function of a clique $c$.

As an example, consider a clique for nodes 3 and 4 as well as link $(3, 4)$ shown in Figure 5. The corresponding potential is a collection of related clique potentials, i.e., $\sum_{c \in C_{34}} \psi_c(\boldsymbol{\sigma}, \boldsymbol{X}) = \left[\frac{(X_3 - X_3(0))^2}{2\sigma^2} + \zeta \sum_{j \in \{2,4\}} h(X_3, X_j)\right] + \left[\frac{(X_4 - X_4(0))^2}{2\sigma^2} \zeta \sum_{j \in \{3,5\}} h(X_4, X_j)\right]$ $+ \alpha_{34} + \sum_{mn \in \sigma_{N_{34}^I}} (\alpha_{34,mn} + \alpha_{mn,34}) \frac{\sigma_{mn}+1}{2}$, where $\sigma_{N_{34}^I} = \{\sigma_{12}, \sigma_{21}, \sigma_{23}, \sigma_{32}, \sigma_{43}, \sigma_{45}, \sigma_{54}, \sigma_{56}, \sigma_{65}\}$ corresponds to the set of neighboring dipoles of $\sigma_{34}$ within the interference range.

The solid line connecting $\sigma_{34}$ and $\sigma_{45}$ indicates the spatial dependence due to the channel contention. The dash line connecting $\sigma_{34}$ with either $X_5$ or $X_6$ indicates the dependence of dipole $\sigma_{34}$ with positions ($X_5$,$X_6$) of a neighboring dipole $\sigma_{56}$. In general, a clique is determined by the interference range.

## *2.5 Optimality and Complexity of Local Model*

When is the local model a good approximation of the global model? In this section, we derive near-optimality conditions for a good approximation. The conditions can be obtained through the approximation error of a local model, communication complexity, and their trade-offs.

### 2.5.1 Communication Complexity

The neighborhood size in a Markov Random Field determines the range of information exchange of a node with its neighbors, and can thus be characterized as communication complexity. The communication complexity of an active dipole can be regarded as the maximum number of active dipoles within its interference range. The number of active dipoles within the interference range is random. Hence we use a deterministic bound for the number of active dipoles for ease of analysis.

Assume that an active dipole satisfies the SINR requirement, i.e., $\text{SINR}_{ij} = \frac{P_i l_{ij}^{-\alpha}}{N_b + \sum_{mn \neq ij} P_m l_{mj}^{-\alpha} \frac{\sigma_{mn}+1}{2}} \geq \text{SINR}_{th}$. Consider a circle centered at the receiver $X_j$ of an active dipole $\sigma_{ij}$ within which there cannot exist any active dipoles for the SINR requirement to hold. The radius of the circle is the SINR contention range for node $j$. Denote the minimum contention range for all active dipoles as $r_c$ which is the minimum distance between any two active dipoles in the network. Only one dipole can be active within a contention range. Now consider interference range outside the contention range where multiple dipoles can be active concurrently, resulting in interference. We now bound the interference region using a circular region of radius $r_f$. This includes a set of active dipoles outside $r_c$ but within $r_f$ as shown in Figure 6. Note that the actual interference range of a receiver is not symmetrical in all directions but bounded by the circular region. This circular region is now considered to be the relevant interference neighborhood for node $j$.

By packing the circular region with small circles of radius $r_c$, we can obtain the

maximum number of active dipoles in the interference neighborhood.

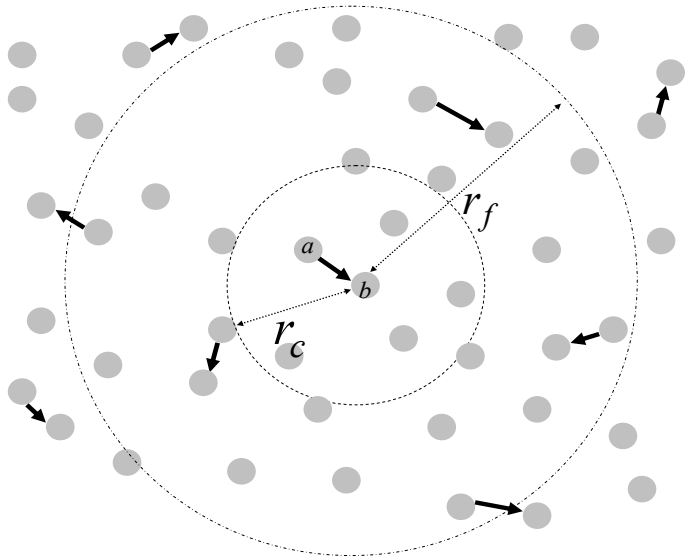



**Figure 6:** Contention range $r_c$ and interference range $r_f$ of an active dipole

Definition 3. *Communication Complexity $\mathcal{C}$: The communication complexity of a dipole $\sigma_{ij}$ is defined as the maximum number of active dipoles within the interference range, i.e., $\mathcal{C} = (\frac{r_f}{r_c})^2$, for $1 \leq i, j \leq N$, $i \neq j$.*

### 2.5.2 Near-Optimality Conditions

We now derive sufficient conditions for a local model to be a good approximation of the global model. For feasibility of analysis, we consider a homogeneous network specified below.

Theorem 1: *Consider a network where $N$ nodes are uniformly distributed, transmit at the same power level ($P_t > 0$), have the same desired SINR threshold ($SINR_{th}$) and the same circular interference range ($r_f$). The network also satisfies the 1-connectivity, where $|\frac{l_{ij} - l_{th}}{l_{th}}| \leq \epsilon_0$ for every $\theta$ radian for $i \neq j$ and $1 \leq i, j \leq N$. Let $\alpha$ be the power attenuation factor of the channel. The average approximation*

*error can be bounded as* $E[\Delta] \leq \epsilon_{\Delta}$, *where*

$$\epsilon_{\Delta} = \begin{cases} \frac{\mathcal{I}}{r_c^2}\left\{(2+\ln\mathcal{C})^2 + \frac{4\pi}{l_{th}}(r_c + \sqrt{N l_{th} r_c})\right\}, & \alpha = 2 \\ \frac{\mathcal{I}}{r_c^2}\left\{\frac{2(2-\frac{1}{\mathcal{C}})^2}{r_c^2} + \frac{4\pi}{l_{th}^2}(\frac{1}{\sqrt{\mathcal{C}}} + \ln(1+\sqrt{\frac{N l_{th}}{\mathcal{C} r_c}}))\right\}, & \alpha = 4 \\ \frac{\mathcal{I}}{r_c^2}\left\{\frac{2(\alpha-2\mathcal{C}^{\frac{2-\alpha}{2}})^2}{(\alpha-2)^2 r_c^{\alpha-2}} + \frac{4\pi}{l_{th}^{\frac{\alpha}{2}}}\left[\mathcal{C}^{\frac{2-\alpha}{4}} r_c^{\frac{4-\alpha}{2}} + \frac{2}{4-\alpha}\right.\right. & \\ \left.\left.(-(\sqrt{\mathcal{C}} r_c)^{\frac{4-\alpha}{2}} + (\sqrt{\mathcal{C}} r_c + \sqrt{N l_{th} r_c})^{\frac{4-\alpha}{2}})\right]\right\}, & else; \end{cases}$$

*with* $\mathcal{I} = 2 l_{th}^{\frac{\alpha}{2}} / \left(l_{th}^{\frac{-\alpha}{2}} - \sqrt{l_{th}^{-\alpha}/SINR_{th} - N_b/P_t}\right)$.

The proof is given in Appendix A. Using the upper bound of the approximation error $\epsilon_{\Delta}$, we can obtain a sufficient condition on the density of active dipoles so that $(\hat{\boldsymbol{\sigma}}, \hat{\boldsymbol{X}})$ is near-optimal for a large network.

Theorem 2: *Let* $\epsilon$ *be a desired performance bound. Assume* $N \gg 1$. *We have* $\epsilon_{\Delta} \leq \epsilon$ *if*

$$r_c \geq \begin{cases} a_1 N^{1/3} + o(), & \alpha = 2 \\ a_2 \sqrt{\ln N} + o(), & \alpha = 4 \\ a_3 N^{\frac{4-\alpha}{4+\alpha}} + o(), & else, \end{cases}$$

*where* $a_1 = \left(\frac{4\pi\mathcal{I}}{\epsilon\sqrt{l_{th}}}\right)^{2/3}$, $a_2 = \left(\frac{2\pi\mathcal{I}}{\epsilon l_{th}^2}\right)^{1/2}$, $a_3 = \left(\frac{4\pi\mathcal{I}}{\epsilon l_{th}^{\frac{4+3\alpha}{2}}}\right)^{\frac{4}{4+\alpha}}$, *and* $o()$ *represents a smaller order term.*

The proof can be obtained through simple algebraic manipulations from Theorem 1, and is thus omitted[7].

[7]From Theorem 1, $\mathcal{C}$ is a function of $r_c$; thus, $\mathcal{C}$ is replaced with $r_f^2/r_c^2$.

### 2.5.3 Trade-Off between Near-Optimality and Complexity

Shown by the above theorems, four parameters impact the approximation error and complexity: channel condition $\alpha$, contention range $r_c$, communication complexity $\mathcal{C}$, and network size $N$.

**Channel, contention range and network size:** First consider large interference where the power attenuation factor $\alpha$ is small. This corresponds to such channel environments as free space $\alpha = 2$, obstructed areas in factories $\alpha \in \{2 \sim 3\}$, and urban areas $\alpha \in \{2.7 \sim 3.5\}$. Theorem 2 shows that when $\alpha = 2$, $r_c \geq O(\sqrt[3]{N})$. This implies that neighborhood size $r_f$ of a node is at least $O(\sqrt[3]{N})$ also. Such a neighborhood size suggests that a node needs to communicate with more and more neighbors when the network size increases. Hence the local model has a large communication complexity. Meanwhile, the maximum number of activated dipoles is $O(N^{2/3})$, i.e., showing a sparsely activated network.

Figure 7 plots the upper bound of the approximation error and the communication complexity for $\text{SINR}_{th} = 20$, $N_b = 0.1$, $r_c = 10$, $r_f \in \{20 \sim 100\}$, $N = 1000$, and $\alpha \in \{2 \sim 6\}$. Nodes are assumed to be uniformly distributed and the distance between two neighboring nodes is chosen to be $l_{th} = 2\ meter$[8]. The large interference corresponds to the flat region in Figure 7, where the SINR requirement is violated, and the upper bound of approximation error $\epsilon_\Delta$ is truncated to remain flat for illustration.

Now consider small interference for a large $\alpha$. This corresponds to such channel environments as shadowed urban areas ($\alpha \in \{3 \sim 5\}$), and obstructed regions in buildings ($\alpha \in \{4 \sim 6\}$) [81]. For $4 < \alpha \leq 6$, $r_c \geq a_3 + o()$. This implies that the neighborhood size remains constant when $N$ increases. Therefore, the Markov

[8]This corresponds to sensor networks for habitat monitoring, battlefield surveillance, and mechanical measurement and monitoring.

Random Field has a meaningful small neighborhood and is an efficient approximation to the global model. Meanwhile, the network has densely activated dipoles, i.e., the number of active nodes can be $O(N)$, showing a densely activated network.

The upper bound of the approximation error is now small, e.g., less than 10% for $\alpha = 4$ and below 1% for $\alpha = 6$ as shown in Figure 7.

**Trade-off:** Figure 7 shows the trade-off between the approximation error and the communication complexity, which is normalized, for $\alpha = 4$, where a smaller $\mathcal{C}$ corresponds to a larger approximation error. The intersection $\mathcal{C}$ and the approximation error, e.g. between the two thick lines, corresponds to an optimal neighborhood with $\mathcal{C}$=16. This corresponds to [9] $r_f = 40$ and $r_c = 10$. In general, for a given $r_f$, $r_c$ can be adjusted to vary $\mathcal{C}$ so that a proper trade-off can be obtained. A wide range of 3 $\sim$ 20 hops for $r_c$ would the most of the feasible scenarios.

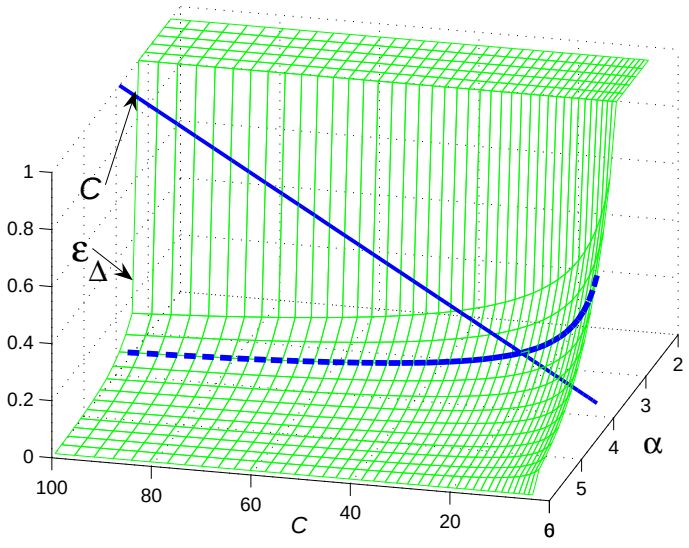



**Figure 7:** Trade-off between performance and complexity

[9] In general, an actual optimal value of $r_f$ depends on a constant that weights the relative importance of performance and complexity.

**Topology**: For the uniform network assumed in the theorem, $l_{th}$, the inter-distance between two neighboring nodes, characterizes the physical topology. Theorem 2 shows that $r_c$ increases with respect to $l_{th}^{1/3}$ and $l_{th}$ for $\alpha = 2$ and $2 < \alpha \leq 4$, respectively. This shows that the rate of growth of the contention range with respect to the inter-node distance.

## *2.6 Distributed Algorithm for Self-Configuration*

The significance of the local model (cross-layer Markov Random Field) is that it allows a spectrum of distributed algorithms to be applied for self-configuration where nodes make decisions using local information from neighbors.

### 2.6.1 Distributed Algorithm

A distributed algorithm obtains a near-optimal configuration by maximizing the approximated likelihood function

$$(\hat{\boldsymbol{\sigma}}, \hat{\boldsymbol{X}}) \quad = \quad \arg\max_{(\boldsymbol{\sigma}, \boldsymbol{X})} P^l(\boldsymbol{\sigma}, \boldsymbol{X}). \tag{18}$$

Due to the spatial Markov property, $P^l(\boldsymbol{\sigma}, \boldsymbol{X})$ is factorizable over cliques. Therefore, maximizing the entire likelihood function reduces to maximizing the localized probability density functions at cliques, i.e., for $1 \leq i, j \leq N$,

$$(\hat{\sigma_{ij}}, \hat{X}_i) = \arg\max_{(\sigma_{ij}, X_i)} P^l(\sigma_{ij}, X_i | X_{N_i}, \sigma_{N_{ij}}), \tag{19}$$

where $P^l(\sigma_{ij}, X_i | X_{N_i}, \sigma_{N_{ij}}) = g_{ij}(\boldsymbol{\sigma}, \boldsymbol{X})$, $X_{N_i}$ and $\sigma_{N_{ij}^I}$ are in the neighborhood of node $i$ and dipole $\sigma_{ij}$. These localized probability density functions are composed of the neighboring nodes and dipoles, and thus the configuration can be updated locally. In addition, the local maximizations result in coupled equations due to the nested Markov dependence among random variables of $\boldsymbol{\sigma}$ and $\boldsymbol{X}$. This shows the need of information exchange among neighbors.

Many algorithms can be used to maximize the localized probability density functions, e.g., stochastic relaxation [10] and message passing [55]. Stochastic relaxation

can avoid local minima [33], and is thus considered in this work. The algorithm has been shown to converge to the global maximum of $P^l(\boldsymbol{\sigma}, \boldsymbol{X})$, asymptotically with probability one [33].

In particular, stochastic relaxation is applied for each node to make local decisions on its new position and transmission activity. Let $\hat{X}_i(t+1)$ and $\hat{\sigma}_{ij}(t+1)$ denote the new position that node $i$ should move to and the activity of link $(i,\ j)$ at time $t+1$, respectively. The distributed stochastic algorithm for $\hat{X}_i(t)$ and $\hat{\sigma}_{ij}(t)$ for $t \geq 1$ is described as follows:

(a) Let $X_{N_i}$ be the set of neighboring random variables of $X_i$ for $1 \leq i \leq N$. Then $\hat{X}_i(t+1) = x_i$, with probability $P^l\left(X_i(t+1) = x_i | X_{N_i}(t)\right) = \frac{\exp(-\psi_i(x_i)/T(t+1))}{\sum_{\forall x_i} \exp(-\psi_i(x_i)/T(t+1))}$, where $\psi_i(x_i) = \frac{(x_i - X_{i0})^2}{2\sigma^2} + \zeta \sum_{j \in N_i^\theta} h(x_i, X_j(t))$.

(b) For $\sigma_{ij}$, $1 \leq i, j \leq N$ and $i \neq j$, $\hat{\sigma}_{ij}(t+1) = \sigma_{ij}$,
with probability $P^l\left(\sigma_{ij}(t+1) = \sigma_{ij} | X_{N_i}(t), \sigma_{N_{ij}}(t)\right) = \frac{\exp(-\psi_{ij}(\sigma_{ij})/T(t+1))}{\sum_{\forall \sigma_{ij}} \exp(-\psi_{ij}(\sigma_{ij})/T(t+1))}$, where $\psi_{ij}(\sigma_{ij}) = \left(\alpha_{ij} + \sum_{mn \in N_{ij}^I} (\alpha_{ij,mn} + \alpha_{mn,ij}) \frac{\sigma_{mn}(t)+1}{2}\right) \frac{\sigma_{ij}+1}{2}$.

Temperature $T$ is now used as a computational parameter, i.e., a cooling constant in the algorithm. Let $T(t) = T_0/\log(1+t)$ with $T_0$=3 be the cooling scheduler [10][63]. This allows an almost-sure convergence to the global minimum Hamiltonian (see [33] for more details).

### 2.6.2 Example

Consider an example how each node updates its position based on stochastic relaxation. An initial topology at $t = 0$ is given in Figure 8. At the first time epoch $(t = 1)$, node 1 has three neighbors $N_1 = \{2, 3, 4\}$, where the neighborhood $N_1$ is

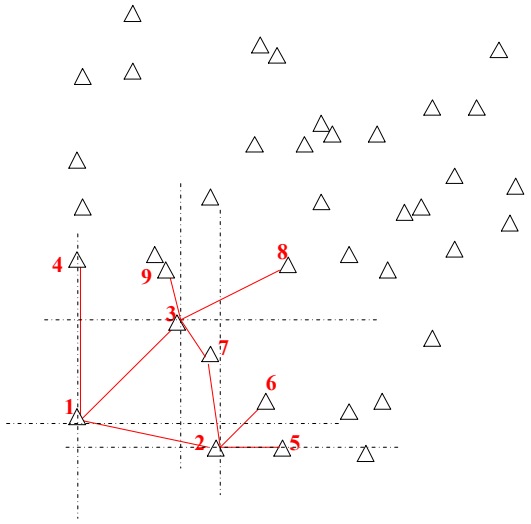



**Figure 8:** Stochastic relaxation for randomly positioned nodes

determined based on the definition of $N_1^\theta$ in Eq.(8). Node 1 then computes the conditional probability given the neighbor positions, i.e., $P(X_1|X_2, X_3, X_4)$. An estimated new position of node $i$, i.e., $\hat{X}_1$, can be obtained as

$$\hat{X_1}(t=1) = \arg\max_{x_1} \; P(X_1 = x_1 | X_{N_1} = x_{N_1}(0)), \tag{20}$$

where $X_{N_1}$=$\{X_2, X_3, X_4\}$ and $x_{N_1}(0) = \{x_2(0), x_3(0), x_4(0)\}$; and "(0)" indicates $t = 0$.

Node 1 then sends its updated position information to the neighbors. This procedure is applied to the rest of nodes and repeated until an equilibrium state is reached.

Link activities $\sigma_{ij}$'s can be updated similarly. Assume that a 1-connected topology is already formed. Stochastic scheduler determines transmissions of nodes in time so that the maximum spatial channel-reuse can be achieved. Meanwhile, the scheduler satisfies the constraints on fairness, 1-connectivity, and SINR [31] [15]. Such a stochastic scheduling makes an efficient use of network resources especially for densely populated wireless sensors/nodes.

Consider multiple time slots which result in a cycle [15]. At the $k$-th time slot of

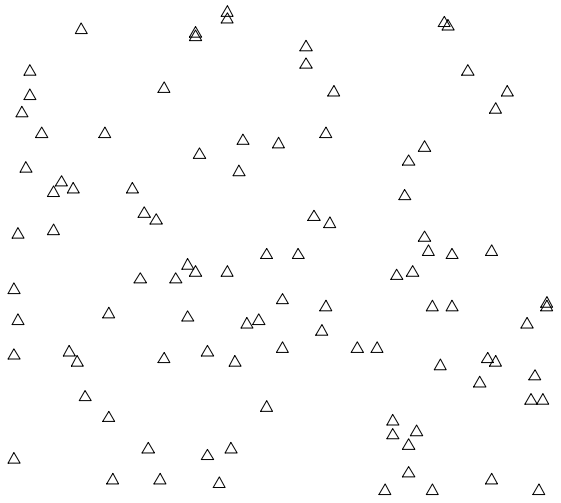


**Figure 9:** Initial Random Configuration: $\boldsymbol{X_0}$

a cycle ($k \geq 1$), an optimal configuration corresponds to the one that maximizes the spatial channel-reuse of unassigned links while satisfying the constraints. To maintain a simple fairness criterion, the active links cannot access the shared channel again until all their neighboring links have accessed the channel.

A time-slot is allocated only based on the spatial dependence among neighboring nodes/links. Hence the stochastic scheduling warrants a distributed implementation. For example, suppose $(\sigma_{38}, \sigma_{26})$ are two neighbors of $\sigma_{14}$. Then the value of $\sigma_{14}$ is determined through the conditional probability, i.e.,

$$\sigma_{14} = \arg \max_{\{-1,1\}} P(\sigma_{14} | \sigma_{38}, \sigma_{26}). \tag{21}$$

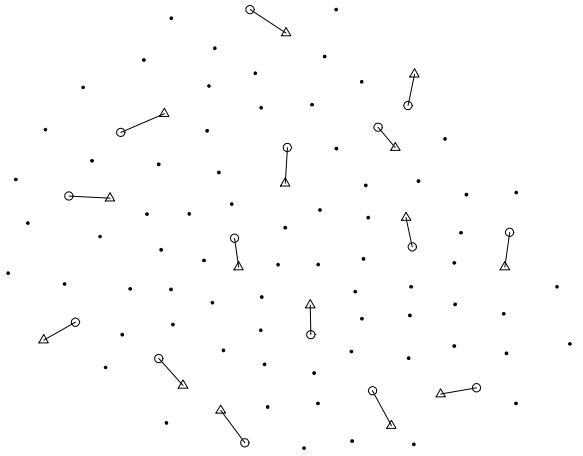
If the resulting $\sigma_{14} = 1$, the directional link $(1, 4)$ gains the channel access.

### 2.6.3 Information Exchange

At time $t$, each node broadcasts its position and adjacent link status to the neighbors: for a node $i$, $(X_i(t), \sigma_{ij}(t) = 1)$ if an adjacent link $(i, j)$ is active; $(X_i(t), \emptyset)$, otherwise. Note that transmission powers are assumed to be fixed as $P_t$.

At time $t+1$, a node $i$ uses such information received from neighbors, i.e., $(X_m(t),$

(a) $\boldsymbol{X}$



(b) $\boldsymbol{\sigma}$ given $\boldsymbol{X}$

**Figure 10:** Self-configuration with localized algorithm

$\sigma_{mn}(t) = 1$) for $m \in N_i$, to update its own local configuration, which is both $X_i(t+1)$ and $\{\sigma_{ij}(t+1)\}$. Here $\{\sigma_{ij}(t+1)\}$ denotes the set of adjacent links of node $i$. For a node $m$ that receives a message ($X_i(t)$, $\sigma_{ij}(t)$=1) from a neighboring node $i$, it relays the message to its neighbors [10].

This bears a similar spirit to that for Bellman-Ford routing algorithm [59]. Hence the near-optimal distributed algorithm uses geo-location information and link activities from neighbors. Note that neighbor positions can be also sensed at a node to

[10]From implementation standpoint, as the stochastic relaxation is robust, convergence may still be guaranteed with delayed information from neighbors [10].

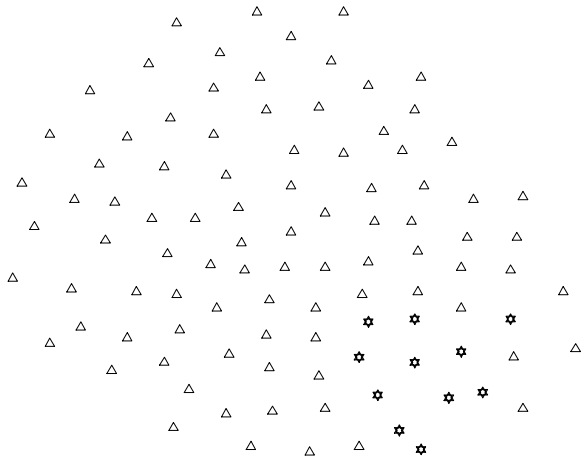


**Figure 11:** Random failure of nodes, marked with stars

avoid information exchange among neighbors.

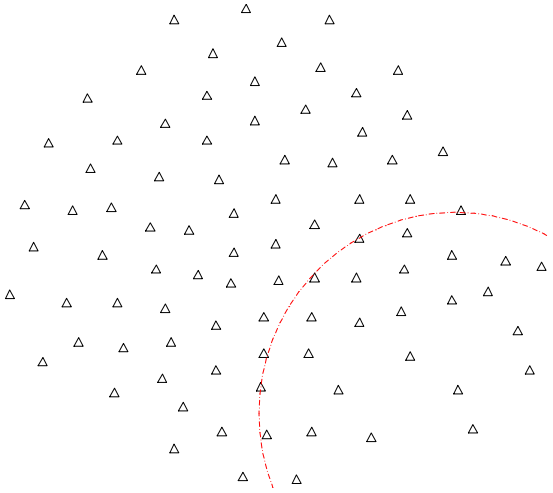
## *2.7 Self-Configuration: Example and Performance Evaluation*

The distributed algorithm is applied to self-configuration of multi-hop wireless networks: (a) 1-connectivity formation of physical topology; (b) scheduling for a logical configuration; and (c) localized failure adaptation.

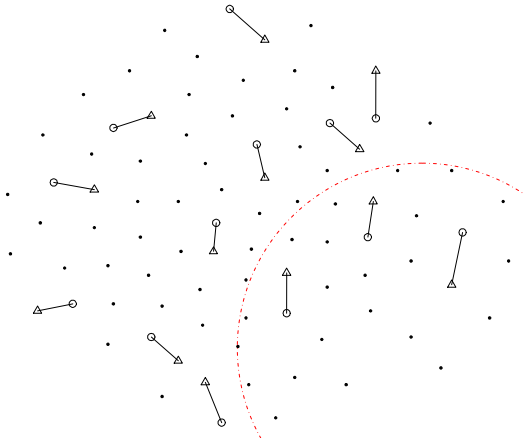
### 2.7.1 Simulation Setup

We select the following parameters in our simulation: network size $N = 100$, an initial configuration ($\boldsymbol{\sigma_0}$=$\mathbf{0}$, $\boldsymbol{X_0}$) as in Figure 9, the threshold of on inter-node distance $l_{th} = 2$ and $\theta = \frac{\pi}{2}$ for the 1-connectivity of the physical topology, $\alpha = 4$ for the channel, and $\text{SINR}_{th} = 20$.

We first obtain a physical topology to ensure the 1-connectivity. Each node updates its position based on the iterative statistical local rules using information from its neighbors, as given in Section 2.6.1. The iteration stops when a steady state is reached, e.g., $|\frac{|X_i - X_j| - l_{th}}{l_{th}}| < 0.01$ for $\forall i$ and $j \in N_i^{\theta}$. Similarly, the logical configuration is obtained by the statistical local rules in Section 2.6.1 given the physical



(a) Localized recovery of $\boldsymbol{X}$



(b) Localized recovery of $\boldsymbol{\sigma}$

**Figure 12:** Localized recovery from random node failures

configuration.

### 2.7.2 Example of Self-Configuration

Figure 9 shows a randomly generated initial configuration. Figure 10 (a) shows a resulting physical topology that forms a 1-connected topology. The physical topology remains fixed during the logical topology configuration. Figure 10 (b) illustrates the resulting logical configuration of the 1-st time slot of a cycle.

Now, consider some nodes fail in a stable network configuration such as in Figure 10 (b). Upon failures, the closest neighbors of failed nodes first get involved in self-recovery by adjusting their positions and selecting other nodes for transmission. This may cause adjustments to the neighbors' neighbors, resulting in cascading changes across the entire network. This means that small perturbations in the network can cause incessant changes to the entire network configuration. Thus, localizing the random failure event is important. Additional penalty terms can be introduced and added in the system potential energy, which serves as reconfiguration costs: $\xi \cdot |(\boldsymbol{\sigma}, \boldsymbol{X}) - (\boldsymbol{\sigma}_s, \boldsymbol{X}_s)|$, where $(\boldsymbol{\sigma}_s, \boldsymbol{X}_s)$ denotes the steady-state network configuration, and $\xi$ is a positive weighting constant that characterizes the cost of change in node positions and/or link activities. Such a constraint would localize the change.

Figure 11 shows a random failure event of wireless nodes. Failed nodes are marked with stars. Localized recovery of the physical topology is shown in Figure 12 (a). The nodes outside the arc are not affected by failures; whereas, the resulting configuration is not globally optimal any more. In a similar way, the failed logical topology can be locally recovered, and shown in Figure 12 (b).

### 2.7.3 Performance Evaluation

We now study the performance, communication complexity and their trade-offs in the simulation setting described in Section 2.7.1. This differs from the analysis in Section 2.5 that is based on uniform topology, provides upper-bounds, and is implemented with ideal distributed management.

**Global and Local Model for the Spatial Dependence Representation:** Figure 13 shows the one-hop capacity, which is achieved based on the global and local model. As a comparison, we also consider the one-hop capacity that is achieved by the

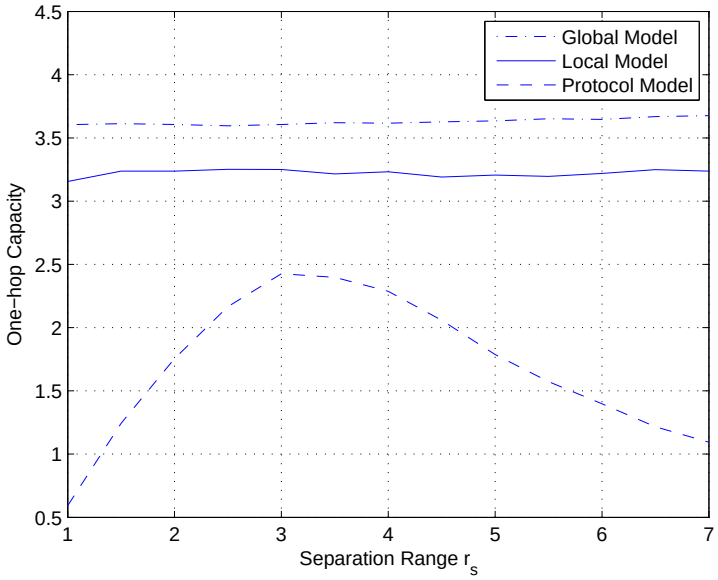



**Figure 13:** One-hop Capacity Comparison: Global, Local, and Protocol Model

protocol (contention) model [31][38][66]. All of these considered algorithms perform in a fully distributed way (i.e., each node makes asynchronous and independent decisions except local information exchange with neighbors).

For this comparison, we consider a wireless network, where total 50 nodes are randomly and uniformly positioned. The interference range of the global model is not limited and covers the entire network; whereas, that of the local model is assumed to be $r_f$=4. For the protocol model, an active link prohibits the other links from accessing the shared channel within a circular range, referred to as the separation range and denoted with $r_s$. Note that the separation range $r_s$ is related only to the protocol model, and both the global and local models are independent of $r_s$.

We consider the range of $r_s \in \{1, 7\}$. Figure 13 shows that for a large separation range $r_s$, the protocol model fails to maximize the spatial channel-reuse. This is because the contention constraint is more harsh than needed. For a small separation range, the protocol model over-utilizes the channel resource, and thus some active links violate the SINR constraint. The net spatial-reuse is decreasing accordingly.

Even with the most optimal separation range (i.e., $r_s$=3), the spatial-reuse of the protocol model is less than that of the local model by 23%. This is because the protocol model cannot consider the non-circular separation of active links (i.e., too simplified neighborhood system of dipoles, defined with a unit circle model). On the contrary, the global and local models consider the non-circular contention of active links and represent the neighborhood system of dipoles more accurately. The spatial-reuse of local model is lower than that of global model by 8% on average. This shows the trade-offed performance for the sake of simplified complexity.

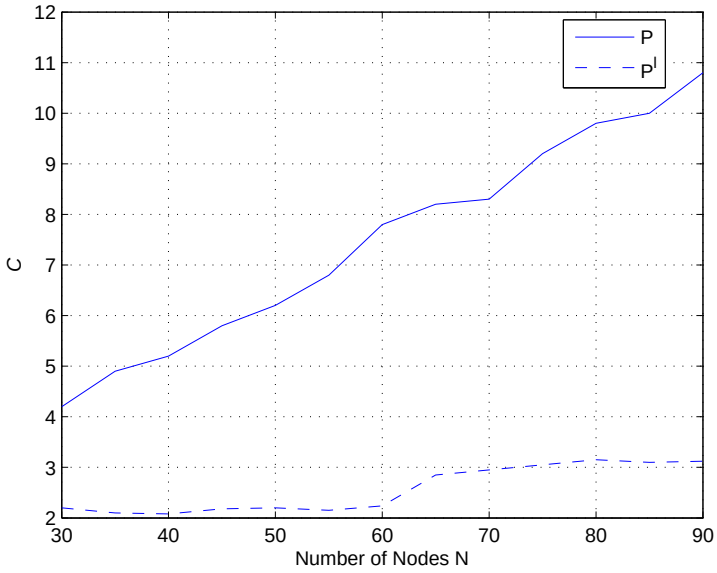
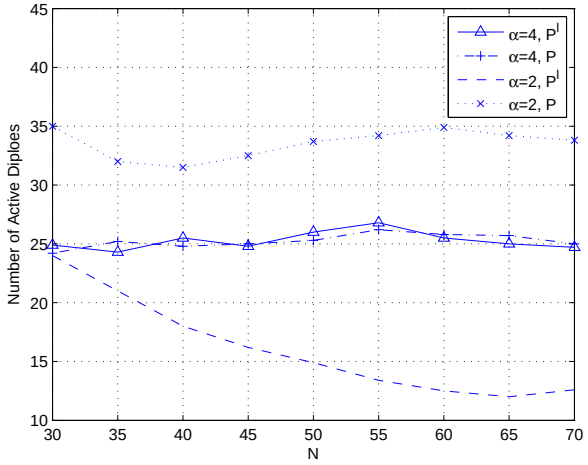


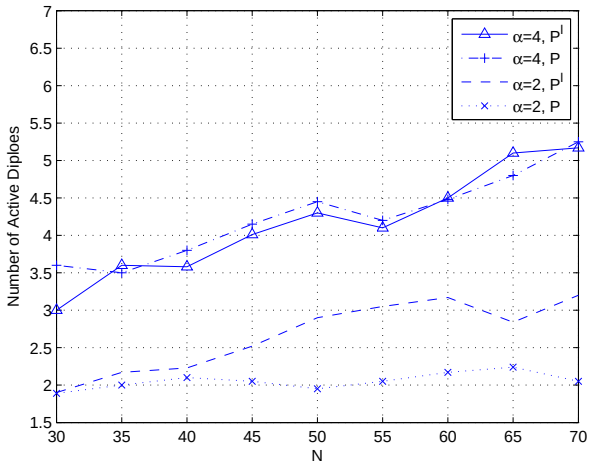

**Figure 14:** Communication complexity $\mathcal{C}$ v.s total number of nodes $N$. $P$ and $P^l$ are the global and local model, respectively.

**Complexity:** We perform 10 independent runs with randomly generated initial physical and logical configurations and the same set of parameters. Figure 14 shows the communication complexity for both the global and local model in terms of network size $N$. The communication complexity $\mathcal{C}$ is obtained by counting and averaging the number of the neighboring active links within the interference range of each active dipole. The results are averaged from 10 experiments with random initial

conditions. $\alpha = 4$ is assumed in the simulations. The communication complexity of the centralized global optimization increases linearly with $N$ since each node needs the information on all the others in the network. The complexity, however, remains bounded for the distributed management as $N$ increases.

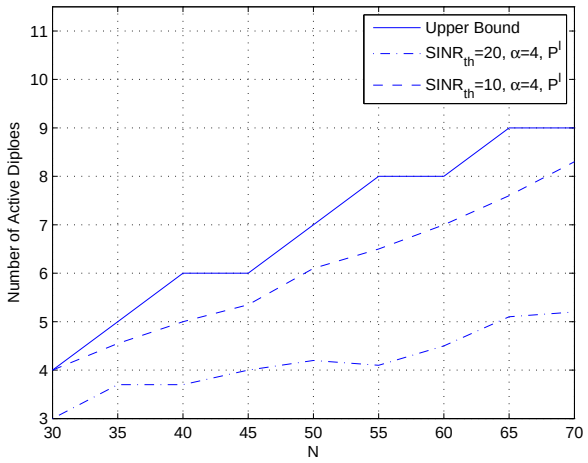



(a) Achieved SINR




(b) One-hop capacity

**Figure 15:** Performance: One-hop Capacity and SINR, where $P^l$ and $P$ are the Local and Global Model, Respectively.




(c) Upper bound of one-hop capacity

**Figure 16:** Upper-bound of One-hop Capacity, where $P^l$ is the Local Model.

**Spatial-Reuse and SINR:** We study further the performance of the distributed algorithm in achieving the SINR requirement and the spatial-reuse maximization.

Figure 15 and 16 show the SINR and the one-hop capacity of the global and local models. The one-hop capacity is measured as the total number of concurrent active dipoles that satisfy the SINR requirements at a time instance [62]. Figure 15 (a) shows that given the neighborhood size of the local model, when the interference decays sufficiently fast, e.g., $\alpha = 4$, the distributed algorithm achieves the SINR requirement. However, in case the interference decays slowly, e.g., $\alpha = 2$, for the same interference range $r_f$, the local model communicates with too few neighbors by ignoring the accumulative interference from far-away nodes, and thus fail to satisfy the SINR requirement. Therefore, depending on the channel condition, a local model may need a larger neighborhood system to account for interference outside the interference range.

Figure 15 (b) shows the one-hop capacity as a function of $N$. For $\alpha = 4$, the resulting logical configuration is near-optimal, where active dipoles are separated

at least $r_c$ distance apart. The corresponding one-hop capacity is $O(\left(\frac{l_{th}(1+\epsilon_0)}{r_c}\right)^2 N)$ resulting from the 1-connected physical topology. The distributed algorithm thus achieves the one-hop capacity at rate $O(N)$, which is the same as that of centralized schemes [38][62] [11].

Furthermore, Figure 16 shows that as $\text{SINR}_{th}$ increases, the spacing $r_c$ between active dipoles increases, and thus the one-hop capacity decreases accordingly.

## *2.8 Conclusion*

In this work, we have developed an analytical framework in which the optimality, approximation and distributed randomized algorithms can be studied for self- configuration. Our findings are as follows.

(a) We have studied the optimality of a network configuration through modeling. We begin with a global model that characterizes the ground truth in regard to assumptions of networks. The ground truth includes internal network characteristics on a wireless channel, a random physical and logical configuration, node decisions, and external management constraints. The model is a Gibbs distribution where the exponent can be regarded as a cost function. This relates the modeling with optimization.

The global probabilistic model characterizes the spatial dependence of node positions and link activities with management constraints.

(b) The complex spatial dependence of the global model can be represented explicitly by a probabilistic graph. The graph provides an approximation, i.e., a local model which is a cross-layer Markov Random Field (MRF) or a random bond model. The near-optimality of a local model is defined when the local model differs from the global model within a given error bound. The complexity of the local model is the

[11] In fact the figure shows that the local model results in a "larger capacity" than the global model due to incorrectly activating more dipoles than $\text{SINR}_{th}$ can tolerate.

communication range of nodes from localized neighborhoods in the Markov Random Field.

(c) Near-optimality conditions for such an approximation are derived under different channel conditions, density of nodes, network size and management requirements. The optimality condition is coupled with the communication complexity which is the total number of interference neighbors in distributed self-configuration. We show that a trade-off exists between the optimality and complexity for the distributed configuration management and depends on different network conditions. Specifically, when the channel has a slow power attenuation of order $2 < \alpha < 4$, the spacing between any two active links (and the communication complexity, i.e., neighborhood system of the local model) should grow at a rate of $O(N^{\frac{4-\alpha}{4+\alpha}})$ with the total number of nodes $N$ in the network. On the contrary, when the channel has a rapid power attenuation of $\alpha > 4$, the spacing between any two active links (and the communication complexity, i.e., neighborhood system) converges to a constant at a rate of $O(N^{\frac{4-\alpha}{4}})$.

(d) We apply the randomized distributed algorithms using the local model. This results in probabilistic inference based on *stochastic relaxation*. The algorithm allows each node to decide its local configuration using only information from neighbors. Local self-configuration collectively achieves a near-optimal global configuration.

(e) We have shown an example of stochastic scheduling and reconfiguration upon failures. We have shown that the distributed algorithm achieves a near-optimal configuration at a bounded communication complexity (i.e., bounded neighborhood system); whereas, the optimal centralized approaches suffer from a large communication complexity which grows linearly with the network size. A disadvantage of such an algorithm is the slow convergence resulting from stochastic relaxation. Other algorithms such as message passing can be used to exploit a faster convergence at the cost of optimality.

From a modeling standpoint, different from top-down approaches which learn a model externally only, we have taken a bottoms-up approach that maps not only external constraints but also network characteristics to a probabilistic model and a dependency graph. As a result, the near-optimality conditions of the local model can be studied through both the internal network properties and external management constraints. One disadvantage of such an approach is that we use simple assumptions which limit the model. Nevertheless, the cross-layer Markov Random Field provides insights on what architecture (neighborhood system) and algorithms can be used for the distributed configuration management of wireless networks. We envision that such a local model can be extended to include more complex and heterogeneous networks. This would constitute some of the future directions.

# CHAPTER III

# MY PREVIOUS WORKS: ON THE COMPLEXITY UPPER BOUND OF THE CONNECTION PREEMPTION PROBLEM IN A MULTI-CLASS NETWORK

## *3.1 Introduction*

The[1] connection preemption problem is well-studied in the multi-protocol label switched (MPLS) networks [93] [2]. On a given route, there will be multiple traffic flows between source and destination. There may be multiple applications and each flow then corresponds to a different connection, e.g., a low-priority file transfer or a high-priority interactive application. In an MPLS network, the flows are kept separate by establishing different label switched paths (LSPs) between the same source and destination.

A new important connection or flow with a high priority may cause existing connections or flows[2] to be preempted. In other words, resources currently in use by one or more existing connections will be allocated to the new connection. The decision on which connections to preempt for the new connection is known in the literature as the connection preemption problem [32] [47] [77] [45]. This decision relies on a priority scheme, which is defined by the preemption priority policy. This policy is one of the resource allocation policies used in MPLS networks. The overall goal of the resource allocation policies is to re-allocate network resources in a more efficient way.

The priority scheme can adopt strict or soft (preemption) priority orders. With

[1]This work is done with Dr. Randal T. Abler and Dr. Ana E. Goulart.

[2]Throughout this paper, the words "connection" and "flow" are used interchangeably.

soft preemption priority orders, a new connection of priority $k$ can preempt a flow with priority $j$ instead of a flow with a lower priority $i$, where $i < j < k$. In this work, a higher priority is always denoted with a larger value. On the other hand, with strict priority orders a new connection always preempts the lowest priority flows first.

In this work, we propose to derive an upper bound of the computation complexity of the connection preemption problem in the prioritized multi-class networks. The derived upper bound provides a guideline about the computational complexity of the connection preemption problem. The preemption problems with both strict and soft (preemption) priority orders are considered.

**Table 2:** Notations on the Preempting Route

| | |
|---|---|
| preemptable flow | flow with a priority lower than $c_{new}$. |
| $N^{c_k}_{n_i,n_{i+1}}$ | number of flows of class $k$ at link ($n_i$,$n_{i+1}$). |
| $N_{n_i,n_{i+1}}$ | number of flows at link ($n_i$,$n_{i+1}$),<br>where $N_{n_i,n_{i+1}} = \sum_{j=1}^{c_{new}-1} N^{c_k}_{n_i,n_{i+1}}$. |
| $N^{c_k}$ | number of flows of class $k$, $N^{c_k} = \sum_{i=1} N^{c_k}_{n_i,n_{i+1}}$. |
| $N^{c_k}_{set}$ | number of feasible combinations of flows of class $k$. |

Although there are many connection preemption algorithms in the literature, the optimality and the complexity upper bound of preemption problem have not been studied in depth yet. For example, Garay and Gopal [32] show that the connection preemption problem is NP-complete, and they present a set of heuristic and centralized preemption algorithms. Peyravian and Kshemkalyani [77] propose two practical algorithms in a decentralized fashion. The problem is better formulated by articulating the decision variables (i.e., the preemption factors): (*i*) the priority of each flow; (*ii*) the bandwidth to be preempted; and (*iii*) the number of flows to be preempted. However, their algorithms are optimal only at the hop level because they do not consider the flows on the other hops.

Additionally, Oliveira [45] represents the connection preemption problem with linear programming and proposes a corresponding adaptive heuristic algorithm. By using the adaptive preemption algorithm, the lower-priority flows can have a chance of minimizing its active bandwidth before being preempted. Anjali [2] formulates the preemption problem in the MPLS networks with a Markov decision process (MDP). Random decisions aimed to reduce the decision time on a preemption problem are proposed in [90]. Other algorithms have been proposed such as a centralized algorithm in [99] and genetic algorithms in [16]. Routing algorithms with knowledge of the preemption problems are considered in [105].

Most of these existing algorithms are very practical but are heuristic in nature. In order to study the optimality and the complexity upper bound of preemption problem, in this work we propose to model the connection preemption problem as a virtual network topology and consider the preemption problem in the domain of a routing problem. That is, we first propose to represent the preemption problem with a graph (i.e., a *virtual topology*), in which a node represents a feasible set of the lower-priority flows and a link represents the set of common flows between two adjacent nodes. Each feasible route in the *virtual topology* will represent a feasible set of flows to be preempted, and the least-cost route can be found with a least-cost routing algorithm [59]. From the complexity of the routing problem, we can obtain that of the preemption problem.

Moreover, we consider the preemption problems with both strict and soft (preemption) priority orders. With soft preemption orders, the preemption problem of multiple service classes cannot be segmented. Thus, the flows of multiple service classes, whose priority order is lower than that of a new connection, are to be considered at the same time. The corresponding graph of the preemption problem with soft preemption orders is called the "multi-layer virtual topology." The computational

complexity is too high to be of any practical use.

However, with strict priority orders, the preemption problem in a multi-class network can be segmented with multiple preemption problems of different priorities. The corresponding virtual topology of each priority is relatively simple.

By constructing (multi-layer) virtual topology, we can derive the computational complexity of both soft- and hard-ordered preemption problems. The routing complexity in the corresponding virtual topology provides a guideline about the upper bound of the computational complexity of the connection preemption problem and the lower bound on the performance.

## *3.2 Graphical Representation of the Connection Preemption Problem*

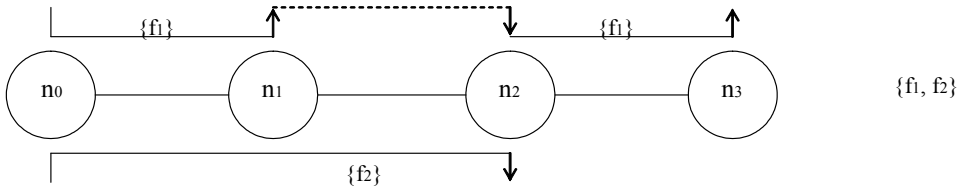
In this section, we show how to represent the preemption problem with a graph. The key idea is to modify the line topology of the preempting route with a mesh flow graph.

The "preempting route" in this work refers to the route for a new connection. For simplicity reasons, we exclude the routing issues from the problem scope. Thus, we assume that the preempting route has already been pre-determined by a proper routing algorithm [32] [77].

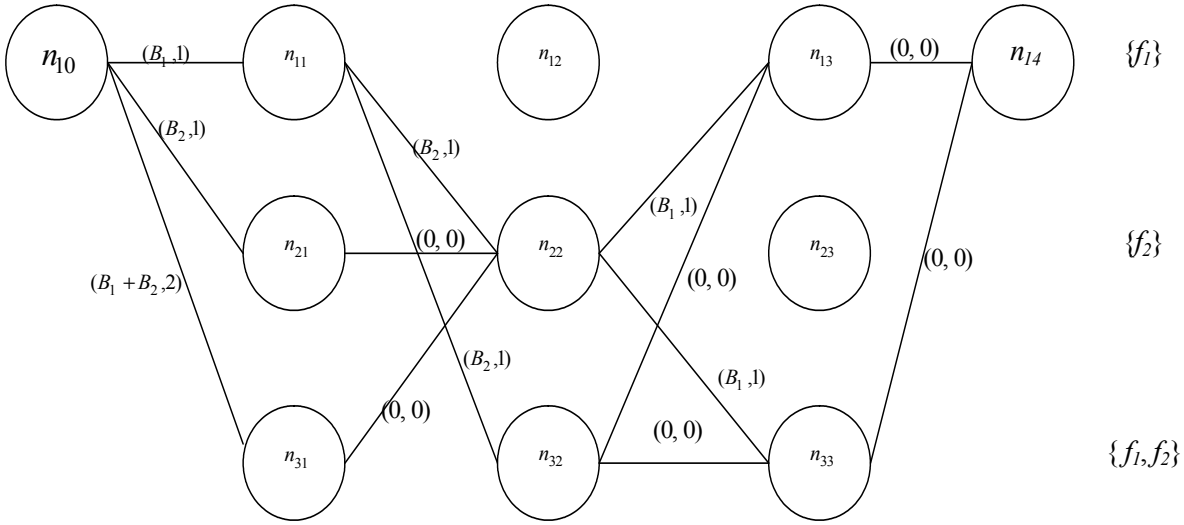
We consider a network with multiple service classes and assume that a new flow belongs to class $c_{new}$ and demands an explicit bandwidth $b_{new}$. Assuming strict priority orders, a preemptable flow is a flow with a priority lower than $c_{new}$.

### 3.2.1 Virtual Topology for Two Service Classes

We first consider a network with only two classes, i.e., class 2 $\gg$ class 1. The unused bandwidth is assumed to be occupied by the virtual traffic of class 0. Let a new flow belong to class 2 (i.e., $c_{new} = 2$) with an explicit bandwidth demand (i.e., $b_{new} > 0$),




(a) A preempting route with two flows, where $b_{new} < B_1, B_2$




(b) Virtual topology

**Figure 17:** Preempting route and the corresponding virtual topology

and let the preempting route $r_1$ consist of $m+1$ nodes (i.e., $n_0,\cdots,n_m$). Figure 17 (a) illustrates the case of flows in route $r_1$, which has four nodes (i.e., $m$=3). The flows have two different classes: class 1 and 2. Considering that the new flow belongs to class 2 (i.e., $c_{new} = 2$), the preemptable flow is the flow with a priority lower than $c_{new}$.

For convenience, some important terminologies are defined in Table 2. For instance, for class 1, the number of flows at link $(n_i, n_{i+1})$ is $N^{c_1}_{n_i,n_{i+1}}$, and the number of flows on the preempting route $r_1$ is $N^{c_1}$. The number of all possible combinations of flows on $r_1$ is $N^{c_1}_{Set}$ (from binomial equation, $N^{c_1}_{Set} \leq 2^{N^{c_1}}$). For example, consider

**Table 3:** Notations for the Link Cost of Virtual Topology

| | |
|---|---|
| $ij$ | node $n_{ij}$ on the virtual topology. |
| $(ij, xy)$ | link $(n_{ij}, n_{xy})$. |
| $\mathcal{L}_{ij,xy}$ | link cost of link $(ij, xy)$. |
| $SF_{ij}^{I}$ | Feasible combination of preemptable flows, which is shard by all nodes on the $i$-th row (i.e., $n_{ij}$ for $1 \leq j \leq m$). |
| $SF_{ij}^{R}$ | Set of all preemptable flows at link $(n_{j-1},n_j)$ for $1 \leq j \leq m$ on the line graph. |
| $N_{ij}^{I}$ | number of flows in the set $\mathrm{SF}_{ij}^{I}$. |
| $B_{ij}^{I}$ | total bandwidth of flows in the set $\mathrm{SF}_{ij}^{I}$. |
| *Fresh-flows* at $(ij, xy)$ | flows in $\mathrm{SF}_{ij}^{I}$, but not in $\mathrm{SF}_{xy}^{I}$. |
| $N_{ij,xy}^{F}$ | number of *Fresh-flows* at link $(ij, xy)$. |
| $B_{ij,xy}^{F}$ | total bandwidth of *Fresh-flows* at link $(ij, xy)$. |

Figure 17. Then, the number of flows of class 1 at each link is $N_{n_0,n_1}^{c_1}$=2, $N_{n_1,n_2}^{c_1}$=1, $N_{n_2,n_3}^{c_1}$=1. The number of flows of any class at each link is $N_{n_0,n_1}$=2, $N_{n_1,n_2}$=1, $N_{n_2,n_3}$=1. Total number of flows of class 1 on the linear topology is $N^{c_1}$=2. Total number of feasible combinations of flows of class 1 is $N_{set}^{c_1}$=3.

Since the highest preempted class (hpc) at a link is either 0 or 1, the preemption problem is now to decide which flows of class 1 to preempt on the preempting route. This preemption problem is defined over a line topology $r_1$. In order to consider all possible combinations of the preemptable flows at each hop of $r_1$, the line graph of $r_1$ is extended to a mesh graph, which is composed of nodes $n_{ij}$, $1 \leq i \leq N_{Set}^{c_1}$ and $1 \leq j \leq m$.

#### *3.2.1.1 Building the virtual topology*

To construct the mesh graph, each node $n_j$ of the preempting route $r_1$, for $1 \leq j \leq m$ (except node $n_0$), is duplicated $N_{Set}^{c_1}$ times. We denote $n_0$ with $n_{1,0}$ instead. And a new node, denoted with $n_{1,m+1}$, is attached at the end of the mesh topology. The resulting mesh graph is called "virtual topology." Figure 17 (b) shows the resulting

virtual topology of the linear topology of Figure 17 (a). For two classes, $N_{set}^{c_1}$=3, thus the linear graph of $r_1$ is extended to the mesh graph composed of 3 nodes for node $n_1$, $n_2$ and $n_3$, with an extra node at the end $n_4$.

The nodes $\{n_{ij}\}$ on the $i$th row in the virtual topology, for $1 \leq j \leq m$, share the same information of a feasible combination of preemptable flows on $r_1$. This shared information is called the imaginary set of flows (SF) to be preempted, and denoted with $\mathrm{SF}_{ij}^{I}$ for node $n_{ij}$.

For example, in Figure 17 (b), $\mathrm{SF}_{ij}^{I}$ is $\{f_1\}$, $\{f_2\}$, or $\{f_1, f_2\}$, where $f_i$ denotes flow $i$. The $\mathrm{SF}_{ij}^{I}$ may include the flows that actually do not traverse the link ($n_{j-1}$,$n_j$) of the preempting route $r_1$. The set of flows that actually traverse the link ($n_{j-1}$, $n_j$) and belong to $\mathrm{SF}_{ij}^{I}$ is called the real set of flows to be preempted and denoted with $\mathrm{SF}_{ij}^{R}$.

Note that for a link $(n_{ij}, n_{xy})$ on the virtual topology, in case $\mathrm{SF}_{xy}^{R} \not\subset \mathrm{SF}_{xy}^{I}$ such as the link $(n_{11}, n_{12})$ in Figure 17 (b), the cost of this link is considered $\infty$. These links whose link cost is $\infty$ are not considered in the routing, which helps reduce the computational complexity.

Since all possible sets of preemptable flows at each hop are represented with the nodes of the virtual topology, each feasible route on the virtual topology corresponds to a feasible set of preemptable flows on $r_1$. Therefore, with a proper routing algorithm, the most efficient route can be obtained, which provides the most optimal set of flows to be preempted.

#### *3.2.1.2 Assigning Costs*

To complete the virtual topology, let us take a closer look on how to define the cost of each link. The link cost of a link is composed of a vector of two elements, i.e., $B_{ij,xy}^{F}$ and $N_{ij,xy}^{F}$ (refer Table 3). For a link $(n_{ij}, n_{xy})$ in Figure 17 (b), the link cost

corresponds to the cost of preempting the set of flows, i.e., { $\mathrm{SF}^I_{xy} - \mathrm{SF}^I_{ij}$ }= the set of *Fresh-flows* at link $(ij, xy)$.

The cost of link $(ij, xy)$ is denoted with $\mathcal{L}_{ij,xy}$, which is

$$\mathcal{L}_{ij,xy} = \begin{cases} (0,0), & \text{if } j = m,\ x = 1,\ y = m+1 \\ (\infty,\infty), & \text{if } |j-y| > 2 \text{ or } y \le j \\ (\infty,\infty), & \text{if } B^I_{ij} \le b_{new} \\ (\infty,\infty), & \text{if } \{\mathrm{SF}^I_{ij},\ \mathrm{SF}^R_{xy}\} \not\subset \mathrm{SF}^I_{xy} \\ (B^F_{ij,xy}, N^F_{ij,xy}), & \text{otherwise,} \end{cases} \tag{22}$$

where $m$ is the total number of links on the line graph, $1 \le i, x \le N^{c_1}_{Set}$, $1 \le j \le m-1$, $1 \le y \le m$, and $\mathrm{SF}^I_{n_0} = \phi$.

The corresponding link cost of a virtual topology can be observed in Figure 17 (b).

Consider the first condition in Eq.(22), which is $\mathcal{L}_{ij,xy} = (0,0)$ for $j = m$ and $y = m+1$. The value $m$ indicates the total number of links on the line graph (e.g., in Figure 17 (a), $m$=3). A link $(j-1, j)$ of the line graph for $1 \le j \le m = 3$ is represented with nodes $n_{ij}$ for $1 \le i \le 3$ in Figure 17 (b). Thus, all links are considered with $n_{ij}$ for $j \le m = 3$ in Figure 17 (b), and the last node $n_{m+1}$=$n_4$ is used for notational convenience only to denote the destination. Thus, $\mathcal{L}_{ix,xy} = (0,0)$ for $j = m$ and $y = m+1$ for a linear topology with $m$ links.

Consider the second condition in Eq.(22), which is $\mathcal{L}_{ij,xy} = (\infty,\infty)$ for $|j-y| > 2$ or $y \le j$. For $|j-y| > 2$, if $\mathcal{L}_{ij,xy} < (\infty,\infty)$, the preemption cost on the intermediate links between $(j-1, j)$ and $(y-1, y)$ would be skipped. To consider the preemption costs of all links on the line graph, we define $\mathcal{L}_{ij,xy} = (\infty,\infty)$ for $|j-y| > 2$. We preempt the flows from link $(n_i, n_{i+1})$ to $(n_{i+1}, n_{i+2})$ for $0 \le i \le m-1$ in the increasing order. If we allow a path between node $n_{ij}$ and $n_{xy}$ on the mesh graph for $y \le j$, there would be a loop on the preemption decisions.

Consider the third condition in Eq.(22), which is $\mathcal{L}_{ij,xy} = (\infty,\infty)$ for $B^I_{ij} < b_{new}$.

This condition implies that the total bandwidth of preemptable flows in set $SF_{ij}^{I}$ is less than the required demand of new flow (i.e., $b_{new}$).

Consider the forth condition in Eq.(22), which is $\mathcal{L}_{ij,xy} = (\infty, \infty)$ for $\{SF_{ij}^{I}, SF_{xy}^{R}\}$ $\not\subset SF_{xy}^{I}$. This condition implies that the considering feasible configuration of preemptable flows on the $i$-th row, i.e., $SF_{ij}^{I}$, includes the flows that neither exist at node $n_{ij}$ in the mesh graph (i.e., at link $(n_{j-1}, n_j)$ on the line graph), nor are preempted at the previous links. For example, the case of the link $(n_{11}, n_{12})$ in Figure 17 (b), $\mathcal{L}_{ij,xy}$ $= \infty$.

Consider the fifth condition in Eq.(22), which is $\mathcal{L}_{ij,xy} = (B_{ij,xy}, N_{ij,xy})$. The cost of $B_{ij,xy}$ and $N_{ij,xy}$ denotes the total bandwidth and total number of preemptable flows in a set $SF_{xy}^{I} \setminus SF_{ij}^{I}$. The set $SF_{xy}^{I} \setminus SF_{ij}^{I}$ denotes a set of preemptable flows that exist at the $x$-th row but not at the $i$-th row.

The vector link cost can be converted into a scalar link cost, which depends on the preemption policy. That is,

$$\mathcal{L}_{ij,xy} = \alpha_B B_{ij,xy}^{F} + \alpha_N N_{ij,xy}^{F}, \tag{23}$$

where $\alpha_B > 0$ and $\alpha_N > 0$. In case a preemption policy declares that the priority of decision variable $B_i$ is higher than that of $N_i$ for $\forall i$ (i.e., $\alpha_B >> \alpha_N$), we first select the least cost path only with the link cost of $B_{ij,xy}^{F}$. If there are more than one path with the same least cost, among the resulting least cost paths, we do another path selection with only the link cost $N_{ij,xy}^{F}$. For the other policies, the least cost path also can be found in a similar way.

Over the final least-cost path, the collection of the information $\mathrm{SF}_{ij}^{I}$ from each node provides the information of which flows to preempt on $r_1$.

### 3.2.2 Validation of the Proposed Method

The flows on $r_i$ (the preempting route of the class $i$) can be classified into three types. The flows that share only a single link with $r_i$ are referred to as type 1 flows, the flows that share more than two consequent links with $r_i$ are referred to as type 2 flows, and the flows that share more than two links inconsequently with $r_i$ are referred to as type 3 flows.

If all flows on $r_i$ (i.e., the preempting route of class $i$) share only a single link with $r_i$, the preemption decision can be done at each hop independently. However, if some flows on $r_i$ share more than one link with $r_i$ (i.e., either type 2 or 3 flows exist), the decision on which flows to preempt at a link may be dependent on that of the other links. This dependence may cause the so-called "merged route problem" to the proposed virtual topology method. The merged route problem is defined that at least two feasible paths share a link with different link costs in the virtual topology.

Figure 17 shows an example of the merged route problem. In the figure, the routes $n_{11}$-$n_{22}$-$n_{13}$, $n_{21}$-$n_{22}$-$n_{13}$, and $n_{11}$-$n_{32}$-$n_{13}$ are referred to as path 1, path 2 and path 3, respectively. For path 1, since the cost of preempting the flow $f_1$ at node $n_{11}$ is already counted at a previous hop, the link cost $\mathcal{L}_{22,13}$ should not include the cost of preempting the flow $f_1$. Thus, $\mathcal{L}_{22,13}$ is null for path 1. However, for path 2, the flow $f_1$ is not preempted at any previous hop, so the link cost $\mathcal{L}_{22,13}$ of path 2 should be $B_1$. Path 1 and 2 require a different link cost at the shared link, which causes the merged route problem.

The path that revisits the same row (of a virtual topology) that it has previously visited is called a returning path, e.g., path 1 in Figure 17 (b). For the merged route problem, the virtual topology method ignores the link cost of the returning paths, represented in Equation (22). For example, $\mathcal{L}_{22,13} = (B_1, 1)$ in Figure 17.

The ignorance of the returning path is actually compensated. That is, since there is always a node containing the information of the preempted flows at the current hop and the previous hops (e.g., $n_{32}$ in Figure 17 (b)), there is always an alternative route that acts for the ignored returning path. The path 3 in Figure 17 is a compensating route for the returning path 1 in this example.

Thus, the cost of all paths can be considered with the proposed method.

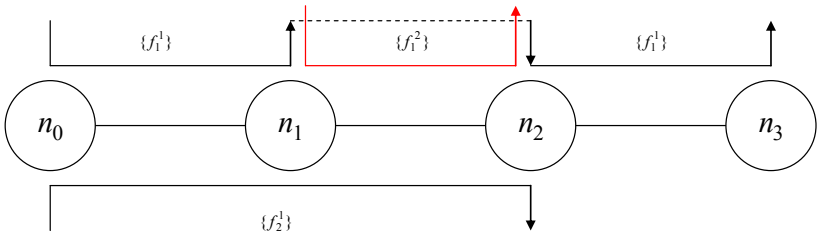



**Figure 18:** Preempting route with four flows, where $b_{new} < B_1^1, B_2^1, B_1^2$

### 3.2.3 Computational Complexity

Since we have converted the preemption problem into the least cost path selection problem on the virtual topology, we derive the complexity of the preemption problem from that of the routing problem. First, note that most nodes in the virtual topology are isolated and without any adjacent links. Thus, the number of nodes that are used in the routing decision is $O(N_{set}^{c_1} \cdot m)$, where $m$ is the number of hops on the preempting route $r_1$, and $N_{set}^{c_1}$ is total number of feasible combinations of flows of class 1. The complexity of constructing the virtual topology of class $c_1$ is in the order of number of links, which is $O((N_{set}^{c_1})^2 \cdot m)$.

In case of using the Dijkstra's shortest-path first algorithm, the complexity of the least cost path selection on the virtual topology is $O((N_{set}^{c_1} \cdot m)^2)$. Therefore, the

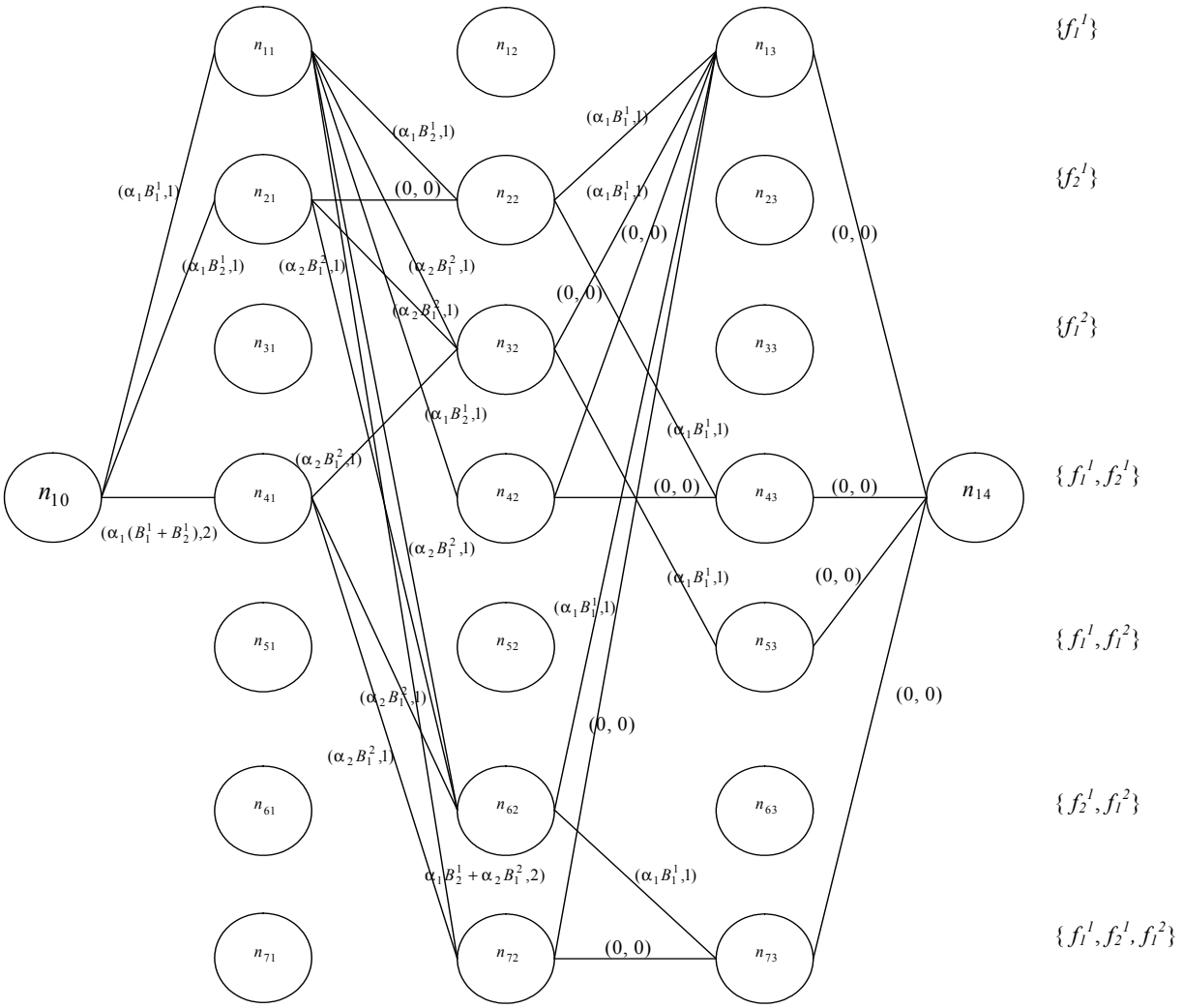



**Figure 19:** Corresponding multi-layer virtual topology of the preempting route

overall complexity (CC) is

$$\begin{aligned} \text{CC} &= O((N_{set}^{c_1})^2 m + (N_{set}^{c_1})^2 m^2) \\ &= O((N_{set}^{c_1})^2 m^2) \\ &= O(2^{2N_{hop}^{c_1}} m^2), \end{aligned} \tag{24}$$

where $N_{hop}^{c_1} = \max\{N_{n_i,n_{i+1}}^{c_1}\}$ for $0 \leq i \leq m-1$.

For comparison, we consider the complexity of the distributed decision with exhaustive searching, which compares all feasible subsets of active flows. Note that at a link $(i, i+1)$ over a preempting route, with distributed preemption decision, the complexity is $\mathrm{O}(2^{N^{c_1}_{n_i,n_{i+1}}})$. Thus, the total complexity on the preempting route results in $O(m \cdot 2^{N^{c_1}_{hop}})$.

Similarly, with the optimal centralized decision with exhaustive searching, the complexity is $O(2^{N^{c_1}} \cdot m \cdot (N^{c_1})^2)$, where $2^{N^{c_1}}$ denotes total number of feasible combinations of the preemptable flows on the preempting route. For a set of flows to be preempted, whose size is bounded by $N^{c_1}$, this set needs to be compared with the set of active flows at each hop over the preempting route $r_1$. At a hop over the route $r_1$, the common flows between this set of flows to be preempted and the set of active flows at the hop correspond to the net flows to be preempted at the hop. The set of these common flows is used to check if the set of preemptable flows can satisfy the requirement of a new flow at each hop over $r_1$. The complexity of this comparison is bounded by $(N^{c_1})^2$ at each hop.

With the proposed method, the complexity is increased by the order of $O(2^{N^{c_1}_{hop}} \cdot m)$ compared to that of the distributed decision; however, it can be reduced by the order of $O(2^{N^{c_1} - 2N^{c_1}_{hop}} \cdot (N^{c_1})^2 \cdot \frac{1}{m})$ compared to that of the exhaustive centralized decision.

## *3.3 Connection preemption Problem with Multiple Service Classes*

In this section, we study if the case of multiple service classes can be represented with multiple independent preemption problems of two service classes. We consider the multi-class networks with both hard- and soft-priority orders.

### 3.3.1 Non-Hardly Ordered Multi-class Networks

For the preemption problem at the softly-order multi-class networks, multiple service classes need to be considered concurrently, which causes additional complexity. For example, Figure 18 shows a preempting route with three service classes for $c_{new} = 3$. On the preempting route in Figure 18, $f_i^j$ denotes a flow $i$ of class $j$ and $B_i^j$ is the bandwidth of the flow $f_i^j$ for $i, j \in \{1, 2\}$.

We represent the active flows of multiple service classes on a linear graph with a multi-layer virtual topology, such as in Figure 19.

The virtual topology includes the flows of class $1,\cdots,c_{new}-1$, thus the resulting computational complexity $CC'$ is

$$\begin{aligned} CC' &= O(2^{2\sum_{k=1}^{c_{new}-1} N_{hop}^{c_k}} m^2) \\ &= O(2^{2c_{new}N'_{hop}} m^2), \end{aligned} \tag{25}$$

where $N_{hop}^{c_k} = \max\{N_{n_i,n_{i+1}}^{c_k} \text{ for } 0 \le i \le m-1\}$[3], and $N'_{hop} = \max\{N_{hop}^{c_k} \text{ for } c_k \le c_{new}-1\}$. The computational complexity is exponential to the total number of flows in the network.

### 3.3.2 Hardly Ordered Multi-class Networks

In the hardly prioritized multi-class networks, a flow of priority $j$ receives better service than a flow of priority $i$ in all cases, for $i < j$. Thus, no active flow of class $j$ at a link will be preempted if there are any active flows of class $i$ at the link, for $i < j$. Let $hpc_l$ denote the highest class of the preempted classes at a link $l$, where $hpc_l < c_{new}$. Thus, we only need to decide which flows of class $hpc_l$ to preempt at the link $l$.

Consider a preempting route with $z$ hops, in which the highest preempted class on

[3] $m$ is total number of hops on the preempting route.

the route is denoted with $hpc$ (i.e., $hpc = max\{hpc_1, hpc_2, \cdots, hpc_z\}$). The preemption decision on class $hpc$ can be done by putting the hops, whose highest preempted class is $hpc$, together side by side, where $1 \leq hpc < c_{new}$. This results in a line topology whose highest preempted class is $hpc$, which is denoted with a route $r_{hpc}$. Now, the preemption decision on $r_{hpc}$ can be done as if there are only two service classes, a preempting class $c_{new}$ and the other preempted class $hpc$, such as in Section 3.2.1.

For example, consider again the preempting route with three service classes in Figure 18. With strict (preemption) priority orders, the highest preempted class at each link of the preempting route is $hpc_1$=1, $hpc_2$=1, and $hpc_3$=1 respectively. As a result, we only need to consider $r_{hpc}$=$r_1$, which is equal to the preempting route in Figure 17 (a). The multi-layer virtual topology in Figure 19 is also reduced into the virtual topology in Figure 17 (b).

To cover the "virtual topology" of class 1 to $c_{new}-1$ over the preempting route, a hierarchical virtual topology that is composed of total $c_{new}-1$ "virtual topologies" will be constructed over the preempting route for the service classes $1 \leq i \leq c_{new}$. With the hardly-ordered priorities, multiple service classes can be considered as multiple independent preemption problems of two service cases. However, are these multiple preemption problems mutually independent? The answer is "NO."

**Parallel/Sequential Computation of $r_i$'s in Hardly Prioritized Networks :** In a network with total $N_C$ service classes, the parallel computation for each $r_i$ is generally infeasible. This is because the decision on $r_i$ depends on the other higher-priority line-topology $r_j$ for $j > i$.

Lemma 1. *For two links $i$ and $j$ on the preempting route, the highest preempted*

*class at link $i$, denoted with $hpc_i$, depends on that at link $j$, denoted with $hpc_j$, for $hpc_i < hpc_j$.*

The proof can be found in Appendix B.

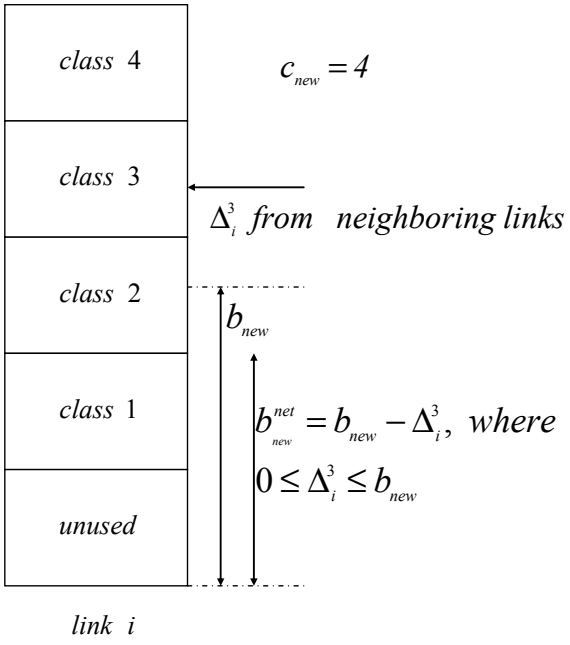



**Figure 20:** Dependence on Neighboring Links

As a result, for an optimal decision, the preemption decision needs to be done sequentially from $r_{c_{new}-1}$ to $r_1$. Depending on the decision at $r_i$, for $j < i$, $r_j$'s need to be re-constructed according to $\Delta_l$ of each link $l$ on the preempting route. Thus, it is infeasible to make a parallel computation of $r_i$'s at the same time, for $1 \le i \le c_{new}-1$.

The resulting overall computational complexity ($\mathrm{CC}''$) is

$$\begin{aligned} \mathrm{CC}'' &= O\Big(\sum_{k=1}^{c_{new}-1} 2^{2N_{hop}^{c_k}} m^2\Big) \\ &= O(c_{new} \cdot 2^{2N'_{hop}} m^2), \end{aligned} \tag{26}$$

where $N'_{hop} = \max\{N_{hop}^{c_k}\}$.

## *3.4 Conclusions*

This work provides an upper bound of the computational complexity of optimal connection preemption decision in the prioritized multi-class networks. We have considered both hardly and non-hardly (i.e. softly) prioritized multi-class networks. In the hardly prioritized multi-class networks, a flow of priority $j$ receives better service than a flow of priority $i$ in any case, for $i < j$.

In the hardly ordered multi-class networks, an upper-bound of computational complexity of the preemption problem is $O(c_{new} \cdot 2^{2N'_{hop}} m^2)$, where $N'_{hop} = \max\{N^{c_k}_{hop}\}$, $N^{c_k}_{hop} = \max\{N^{c_k}_{i,i+1}\}$ for $0 \leq i \leq m-1$, and $N^{c_k}_{i,i+1}$ is the number of flows of class $k$ at link $(i, i+1)$ on the preempting route.

In the non-hardly ordered multi-class networks, an upper-bound is $O(2^{2c_{new}N'_{hop}} m^2)$. The result shows that with the hard-priority orders, the computational complexity can be significantly lowered.

Moreover, with the proposed decision algorithm, in the hardly prioritized multi-class networks, for a class $k$ (or class $c_k$), the computational complexity of preemption problem can be reduced from $O(2^{N^{c_k}} m (N^{c_k})^2)$ of the exhaustive optimal centralized decision to $O(2^{2N^{c_k}_{hop}} m^2)$.

# CHAPTER IV

# DISTRIBUTED MANAGEMENT OF POLICY-BASED RESOURCE ALLOCATION IN MULTI-CLASS NETWORKS

## *4.1 Introduction*

Preemption occurs at a prioritized multi-class network, where a new call needs to be set up with a high priority between a source (S) and destination (D) pair [32] [47] [75] [77]. When the capacity is insufficient at all feasible routes between the S-D pair, some existing flows of the lower priorities need to be preempted to accommodate the new call. Preemption is to decide whether to remove a certain low priority flows to free the reserved bandwidth for the new call at a chosen route [32] [77]. Hence the goal is to decide on whether to remove an active flow so that the total preempted bandwidth can be minimal under given constraints e.g., bandwidth demand of a new call setup; and available free bandwidth at a link [1].

Connection preemption allows a new high-priority connection to access heavily crowded core networks, e.g. multi-protocol label switched (MPLS) networks. Connection preemption also improves resource utilization by allowing low-priority flows to access unused bandwidths [39] [93]. Preemption sees potential applications in emerging networks also. For example, in 802.11e Wireless LAN, delay sensitive IP packets in expedited forwarding (EF) class can be served earlier than the best-effort packets through preemption [76]. Multi-level preemption and precedence (MLPP) has been

[1]The preempted flows are usually rerouted to other paths. Hence preemption and rerouting can be considered jointly with slightly different objectives [93]. This work, however, focuses on preemption on a given path without considering rerouting.

proposed to classify calls by their importance, which can be used for military networks as well as various commercial services [8] [71].

There are two significant challenges for preemption. One is complexity, i.e., preemption is known to be NP-complete [32]. A cause of the computation complexity is a large number of active flows supported by a core network for which preemption decisions need to be made. For example, for a 1Gbps link, if the bandwidth of each flow is in the order of Kbps, there would be thousands of flows supported per link. The other challenge is spatial dependence of decisions at different nodes. A flow generally passes through multiple nodes, which makes it difficult for each node to make preemption decisions independently. That is, preemption is network-centric rather than node centric.

Centralized preemption maintains at a centralized node the location information of active flows, their priorities and bandwidth occupancies at the entire route. The centralized node then decides which active flows to preempt upon the request of a new call. Therefore, centralized preemption is always optimal, resulting in minimal preempted bandwidths. But the amount of management information needed and the associated computation complexity can be overwhelming at the centralized node. For example, let $F_t$ be the total number of distinct flows per priority class at the route. Each flow have two states, preempted or not preempted. The total number of possible states is $O(2^{F_t})$ for making a centralized decision. When $F_t$ is in the order of hundreds or thousands [49], centralized preemption decision becomes computationally intractable. Decentralized preemption is then sought for to reduce the amount of management information and the computational complexity [77].

Decentralized preemption maintains at each node local information, i.e., active flows at the adjacent link, their priorities and bandwidth occupancy. Such information is readily available and local at a node. A node then decides, independent of the other nodes, which connections to preempt. This, however, may cause conflicting

local decisions on the same flows across multiple links, resulting in more preempted bandwidths than necessary. Hence, decentralized preemption neglects the spatial dependence for the flows across multiple links, and may thus perform poorly. But the amount of management information and the computation complexity are greatly reduced.

For example, let $F$ be the maximum number of active flows per link. Let $L$ be the number of hops on the route. The total number of states is $L2^F$ at each link. Since $2^F \ll 2^{Ft}$, compared with centralized preemption, decentralized schemes have a much smaller search space for preemption decisions. Therefore, most algorithms in the literature are on decentralized preemption (see [75] [77] and references there in). Decentralized preemption, however, neglects important spatial dependence for flows across multiple links.

Distributed decisions take into account spatial dependence through local information exchange among neighboring links. In fact, distributed preemption can be considered as a general setting of centralized and decentralized preemption. Centralized preemption corresponds to one extreme with the entire route as the neighborhood whereas decentralized decisions correspond to the other extreme without any neighborhood. Therefore, the communication/computation complexity can be characterized in terms of neighborhood size. There is a trade-off between the optimality and the complexity. A strong requirement for distributed decisions is to achieve a certain optimality requirement, i.e., to be within a given error bound to the optimal performance, at moderate complexity. This implies that a collection of local decisions made at nodes based on the local information and exchange with neighbors should achieve a near-optimal preemption at the network level.

Numerous distributed algorithms and protocols have been developed based on empirical studies [32] [47] [75] [77]. The performance of these algorithms though is

usually tested through simulation. Simulations would not provide quantifiable conditions on *when* and *how* distributed preemption can be nearly optimal within a given error bound. This requires modeling a large number of dependent decision variables of end-end flows and assessing the performance of distributed algorithms accordingly relative to that of the optimal one. In general, it has been shown to be a difficult problem to develop a distributed algorithm whose performance is predictable and within a tolerable degradation from that of the optimal scheme [95]. Hence, the open issues are: (a) *When* can the corresponding distributed decisions collectively result in a near-optimal global preemption? (b) *How* to model a large number of dependent decision variables and to obtain near-optimal local decisions using distributed algorithms?

We apply machine learning approaches to study these issues. How is distributed preemption related to machine learning?

**Machine learning perspective:** A machine learning view of distributed preemption is that individual nodes "learn to make decisions" collectively. Ideally, if each node had complete information on either active flows at a route, the node would be able to make correct decisions on which flows to preempt. However, at a given time, a node has only partial information on active flows and its neighbors' decisions on the flows to preempt. But a node can adapt, i.e., learn to make decisions based on those of its neighbors'. As neighbors learn from neighbors' neighbors, a node would indirectly learn what farther nodes decide but with a delay. Eventually, all nodes would collectively make local decisions, resulting in a near-optimal preemption at the entire route.

How would machine learning benefit preemption? The problem of collective learning and decision making has been a keen interest in machine learning and adaptive

control [10] [34], but has just begun to see applications in networking [51] [64]. Machine learning provides a framework in which a large number of decision variables can be treated jointly. Spatial dependence among these variables poses a key challenge to preemption due to end-end flows, is an origin of high communication and computational complexities, and has not been dealt with in depth in the prior work. Machine learning provides feasible approaches for us to study this problem as summarized below.

(a) *Global model of distributed preemption decisions:* Our first step is to develop a probabilistic model which represents explicitly both the long-range and short-range spatial dependence of distributed preemption decisions over a pre-determined preempting route. The randomness originates from randomly arriving/departing active flows and their locations, and also from the insufficient and inaccurate local information for distributed decisions. Thus, preemption decisions made on flows at each node over a given route are also random. We first characterize a cost function for preemption as a "Hamiltonian" (or system potential energy) [63]. A Hamiltonian combines local preemption decisions, objectives in terms of bandwidth savings, and constraints into a single quantity. The constraints are on link capacity and coherent local decisions for flows across multiple hops. The Hamiltonian is then used to obtain a spatial probabilistic model as a Boltzmann distribution [33].

(b) *Markov Random Field (MRF) and sufficient conditions:* The spatial dependence can be characterized through a probabilistic dependency graph of Graphical Models [33] [53] [55] in machine learning. A probabilistic dependency graph provides a simple yet explicit representation of the spatial dependence among random variables. We show that if the dependence of decision variables in Boltzmann distribution is Markov, a globally optimal preemption decision can be obtained collectively

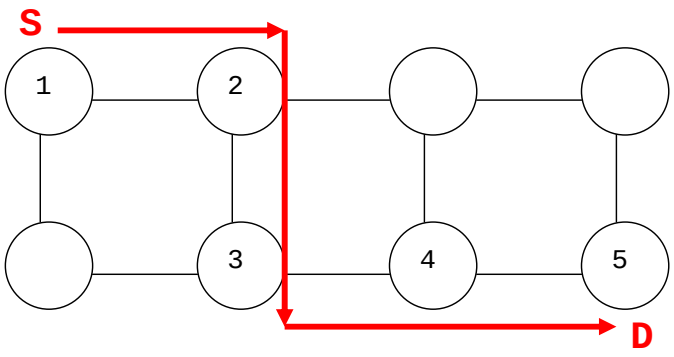
by local decisions through local information exchange with neighbors. Such a Markov probabilistic model is known as a Markov Random Field [33].

In general, distributed decisions may not be spatially Markov but may be well approximated by a Markov Random Field. As the spatial dependence is caused by flows across multiple links, we identify traffic patterns of active flows that result in approximated spatial Markov dependence. We then define the near-optimality of distributed decisions as the distance between the centralized and distributed decisions, measured in the Hamiltonian, and obtain sufficient conditions accordingly (i.e., distance resides within an error bound).

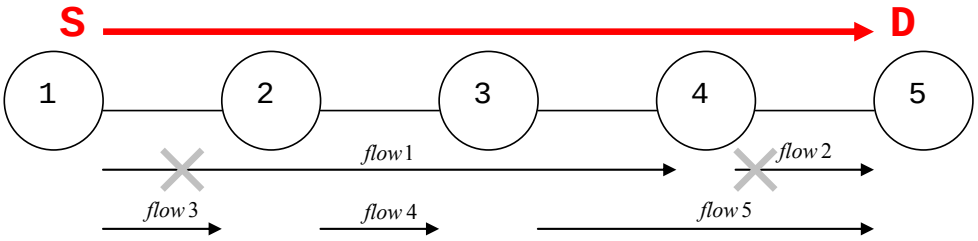
(c) *Distributed Decision Algorithms:* A near-optimal distributed algorithm is derived based on the developed probabilistic network model. The algorithms can be implemented through either message passing [55] or Gibbs sampling [33].

(d) *Trade-offs:* The issues of "when" and "how" are on *performance* and *complexity* of distributed preemption decisions, and their trade-offs. The *performance* measures the optimality of distributed preemption decision relative to that of the centralized decision.

The communication/computation complexity of distributed preemption can be characterized by the amount of information used in decision making. Distributed decisions reduce complexity by using information exchange only with neighbors, but may deviate from the optimal performance. Hence performance and complexity trade-off is to be studied through both analysis and simulation.




(a) A new call arrival into a heavily crowded network




(b) Preemption decisions at each hop over the route of a new connection

**Figure 21:** Example of preemption

## *4.2 Distributed Preemption*

### 4.2.1 Example

Figure 21 (a) shows an example network, and Figure 21 (b) illustrates distributed preemption on a given route. Assume that a new request is made on the route between the SD pair, and every flow has the same bandwidth. The centralized preemption would preempt two existing flows as { flow 1, and flow 2 } that are denoted as $X$. Such a preemption is obviously optimal. Now consider distributed preemption. Take node 2 as an example. The local information available at node 2 includes the priority and the bandwidth of flow 1, 3 and 4 that pass through this node. When the bandwidth is the same for all flows, node 2 may decide to preempt flow 4 without knowing that nodes 1 and 3 both decide to preempt flow 1. Such a decision would result in more flows to be preempted than necessary compared with the centralized decisions. In contrast, flow 1 would be chosen at node 2 if node 2 also has information on the

decisions at the neighbors (i.e., node 1 and 3). The preemption decisions at a node will affect either immediately or after the being-preempted flows are rerouted [74].

This example shows:

(a) Local decisions are spatially dependent. The spatial dependence originates from both flows trespassing multiple nodes and link capacity that constrains flows on a link/node.

(b) The spatial dependence can be taken into consideration by exchanging local decisions with neighbors.

(c) Decisions exchanged would enhance consistent decisions, and thus improve the optimality of local preemption done at nodes.

This motivates the needs of cooperative distributed preemption which is formulated below.

### 4.2.2 Problem Formulation

**Assumptions**: For distributed preemption, we assume that a preempting route (i.e., denoted with $R_{1L}$) is pre-determined for a new connection [32] [77], and composed of nodes 1 to $L$. We assume that the traffic flows on the route belong to multiple priority classes $1, \cdots, i_{max}$, and a new connection belongs to class $i_{new}$ and demands bandwidth $c_{new}$.

**Variables**: Let $S_F$ be a set of all active flows on the route, where $S_F = \{f^1, \cdots, f^{|S_F|}\}$ with $|S_F|$ being the cardinality of set $S_F$. $f^k$ and $B^k$ denote flow $k$ on the preempting route and its bandwidth. Let $\boldsymbol{f_{ij}}$ be the set of all active flows at link $(ij)$.

Let $d_{ij}^k$ denote the preemption decision on flow $k$ at link $(ij)$ for $1 \leq i \leq L-1$ and $j = i+1$. $d_{ij}^k$=1 if link $(ij)$ decides to preempt the flow $k$; $d_{ij}^k$=0, otherwise [2]. Let $\boldsymbol{d_{ij}}$ denote the set of local preemption decisions on all active flows at link $(ij)$.

[2]Preemption decisions at link $(ij)$ are actually done at node $i$. For convenience, the decisions are done at the link $(ij)$ throughout this work.

Let $\boldsymbol{d}$ denote all local decisions at the route, where $\boldsymbol{d} = \{\ \boldsymbol{d_{ij}}, \text{for} \forall i, j\}$.

Let $d_{1L}^k$ denote the preemption decision on flow $k$ over the entire preempting route. That is,

$$d_{1L}^k = 1 - \prod_{i=1}^{L-1}(1 - d_{ii+1}^k), \tag{27}$$

where $d_{ii+1}^k$ is a local decision made at node i (for link (i, i+1)). This means that $d_{1L}^k = 1$ for flow $k$ to be preempted from the given route if at least one node decides to remove the flow, and $d_{1L}^k = 0$ if all nodes decide to keep the flow. Let $\boldsymbol{d_{1L}}$ denote the set of preemption decisions on the flows over the preempting route $1L$.

**Problem statement**: Consider the information maintained at node $i$ ($1 \leq i \leq L-1$): (a) complete local information on active flows at link $(ij)$ which includes flow ID $k$, class priority $i^k$, bandwidth $B^k$, for $k \in \boldsymbol{f_{ij}}$ and $1 \leq k \leq |S_F|$; and (b) neighbor information that includes decisions from the neighboring links within $N_d$ hops.

The constraints of preemption include link capacity and consistent local decisions on the same flows across multiple links. An objective of distributed preemption is to minimize the total preempted bandwidth for accommodating the new call. Distributed preemption is to obtain $d_{ij}^k$ at node $i$ for $1 \leq i \leq L-1$ using the local and neighbor information, where $d_{ij}^k$'s for $1 \leq i \leq L-1$ should collectively achieve the objective under the given constraints. A key challenge is how to model and coordinate a large number of spatially dependent decisions $d_{ij}^k$'s locally to achieve the global objective of preemption.

## *4.3 Probabilistic Spatial Model of Preemption Decisions*

We first develop a global model to represent the spatial dependence of a large number of distributed preemption decisions. We then derive a local model as an approximation. The global and local models are developed through probabilistic graphical models in machine learning.

### 4.3.1 Global Model

A global model should include accurate spatial dependence resulting from objectives and constraints on distributed preemption decisions. The objective function is to minimize total preemption costs and a function of priority of preempted flows, amount of preempted bandwidth, and total number of preempted flows [77] [48].

#### *4.3.1.1 Deterministic Flows*

To model the spatial dependence, we first assume that a set of active flows is given (and thus deterministic). To set up a new connection of a high priority, the objective is to minimize an overall preemption cost $H(\boldsymbol{d})$, where

$$H(\boldsymbol{d}) = \sum_{(i,j),k} \alpha_k B^k d_{ij}^k - \sum_{(i,j),k_1} \sum_{(m,n)\neq(i,j),k_2} \alpha_{k_1} B^{k_1} d_{ij}^{k_1} d_{mn}^{k_2} \delta(k_1,k_2) + \beta \sum_{(i,j)}$$

$$(c_{new} - \sum_k B^k d_{ij}^k - B_{ij}^0)^2 U(c_{new} - \sum_k B^k d_{ij}^k - B_{ij}^0),$$

where $\alpha_k$ is the priority weight of flow $k$, $B_{ij}^0 \geq 0$ is the unused bandwidth at link $(i,j)$, $\delta(k_1,k_2)$=1 if $k_1$=$k_2$; 0, otherwise. $\beta$ is a large positive constant.

The first term is the cost corresponding to the total preempted bandwidth. An objective of preemption is to minimize preempted bandwidth by removing just enough lower-priority flows for accommodating the new call.

The second term corresponds to the constraint on making consistent local decisions on the same flow across multiple links. Consider the term "$\alpha_{k_1} B^{k_1}\ d_{ij}^{k_1} d_{mn}^{k_2} \delta(k_1,k_2)$" as an example. For flow $k$ that uses both links $(i,j)$ and $(m,n)$, this quantity is minimized when these two links decide to preempt the same flow, i.e. $d_{ij}^k$=$d_{mn}^k = 1$.

The third term corresponds to the capacity constraint, where $U(c_{new} - \sum_{(i,j),k} B^k d_{ij}^k - B_{ij}^0)$ is an indicator function. $U(c_{new} - \sum_{(i,j),k} B^k d_{ij}^k - B_{ij}^0) = 0$ if the total preempted bandwidth at link $(i,j)$, $\sum_{(i,j),k} B^k d_{ij}^k + B_{ij}^0 \geq c_{new}$, and $U(c_{new} - \sum_{(i,j),k} B^k d_{ij}^k - B_{ij}^0) =$

1, otherwise.

In (28), by replacing $\alpha_k B^k$ and $\alpha_{k_1} B^{k_1}$ with $(\alpha_k B^k + \gamma_k)$ and $(\alpha_{k_1} B^{k_1} + \gamma_{k_1})$ where $0 \leq \alpha_k$, $\alpha_{k_1}$, $\gamma_k$, $\gamma_{k_1} \leq 1$, we can consider the priority of preempted flows, the amount of preempted bandwidth and the number of preempted flows together. The objective we consider in (28) corresponds to $\gamma_k$=0 and $\gamma_{k_1}$=0 for simplicity.

Given a set of active flows, centralized preemption can always preempt an optimal subset of flows, resulting in minimal preempted bandwidths. The optimal subset of preempted flows can be obtained through deterministic optimization, for example, linear programming [98].

Distributed preemption allows each link iteratively and asynchronously updates its decision based on local information and neighbors' decisions. Each link/node only accesses to limited and initially inaccurate information from near neighbors and missing information from far neighbors. But through neighbor's neighbors, such information would eventually propagate to all nodes, resulting in globally consistent decisions. Such an idea results from dynamic programming and thus in a similar spirit as Bellman-Ford equation [3] [13].

#### *4.3.1.2 Random Flows*

We now extend the above formulation to random flows. What and how many flows are active at which links are related to user behaviors and thus random. Hence active flows and their aggregation at individually links should be regarded as random variables. Preemption decisions made on active flows should be considered as random also. A set of decisions thus form a sample space $S_{\boldsymbol{d}} = \{\boldsymbol{d}\}$, a subset of which consists of events due to distributed decisions. A given set of decisions on a given set of flows

[3] $d_x(y) = \min_{v \in N_x}\{c(x,v) + d_v(y)\}$, where $d_x(y)$ is the cost of least-cost path from $x$ to $y$; $v$ is a neighboring node of node $x$ (i.e., $N_x$); and $c(x,v)$ is the link cost of link $(x,v)$.

is then a sample (realization) of an event. One such sample is given in the example shown in Figure 21, where $\boldsymbol{d}$=$\{d_{12}^1, d_{12}^3, d_{23}^1, d_{23}^4, d_{34}^1, d_{34}^5, d_{45}^2, d_{45}^5\}$= {1, 0, 0, 1, 1, 0, 1, 0}. This relates random and deterministic flows and decisions.

To obtain an optimal set of preempted flows, stochastic rather than deterministic optimization should be used, and this requires a probability distribution of $\boldsymbol{d}$.

Such a probability distributions can be obtained through graphical models defined on neighborhood systems [33]. The neighborhood system can be characterized by Hamiltonian (energy) [33]. The energy of a decision variable results in a per-variable preemption cost, $\alpha_k B^k d_{ij}^k$, in the first terms of (28). Interactions between decision variables result in $\sum_{(i,j),k_1} \sum_{(m,n)\neq(i,j),k_2} \alpha_{k_1} B^{k_1} d_{ij}^{k_1} d_{mn}^{k_2} \delta(k_1, k_2)$ as the second terms of (28).

Combining the cost and the constraints, $H(\boldsymbol{d})$ corresponds to a Hamiltonian of $\boldsymbol{d}$. The Hamiltonian results in a Botlzmann (Gibbs) distribution [33] [63],

$$P(\boldsymbol{d}) \quad = \quad Z_0^{-1} \exp\left(\frac{-H(\boldsymbol{d})}{T}\right), \tag{28}$$

where $T$ is the temperature [33], and $Z_0$ is a normalization constant. Hence the Boltzmann distributed provides a mathematical representation of spatial dependence in distributed decisions. The minimum of the Hamiltonian corresponds to optimal preemption decisions that maximize the probability.

#### *4.3.1.3 Probabilistic Graphical Models*

The intricate spatial dependence among a large number of decision variables can be represented explicitly by probabilistic graphical models. A graphical models relates a probability distribution of random variables with a corresponding dependency graph [33] [53] [55]. A node in the graph represents a random variable and a link between two nodes characterizes their dependence. In particular, a set of random variables $\boldsymbol{v}$ forms Gibbs Random Field (GRF) if it obeys a Gibbs distribution [63]. A Gibbs

distribution has the same form as a Boltzmann distribution. A Gibbs distribution satisfies the positivity condition, meaning that all decisions have a positive probability. One other important property is the spatial Markov property defined by the neighborhood system and shown by Hammersley-Clifford theorem.

Markov Random Fields correspond to an interesting type of probabilistic graphical models where a random variable is conditionally independent of the other nodes given its neighbors. The conditional independence is spatially nested, i.e., a node depends on its far neighbors through neighbors' neighbors. Such nested dependence can be observed explicitly through local connections among nodes in a dependency graph. The corresponding Boltzmann (Gibbs) distribution is thus factorizable over clique potentials [33].

An important implication to preemption is that if distributed decisions result in an MRF, local decisions using neighbor information is optimal. Do distributed preemption decisions $\boldsymbol{d}$ form a Markov Random Field? We visualize the spatial dependence of the decision variables provided by the Gibbs distribution. A factor graph [55] is shown in Figure 22.

In Figure 22, $g_{ij}(\boldsymbol{d})$ is a local function that encompasses the flows passing through link $(i,j)$, i.e.,

$$\begin{aligned} g_{ij}(\boldsymbol{d}) &= \sum_{k\in\boldsymbol{f_{ij}}} \alpha_k B^k d_{ij}^k - \sum_{k_1\in\boldsymbol{f_{ij}}} \sum_{(m,n)\neq(i,j),k_2\in\boldsymbol{f_{mn}}} \\ &\quad \alpha_k B^k d_{ij}^{k_1} d_{mn}^{k_2} \delta(k_1,k_2) + \beta U(c_{new} - \sum_{k\in\boldsymbol{f_{ij}}} B^k d_{ij}^k - B_{ij}^0), \end{aligned} \tag{29}$$

where $\boldsymbol{f_{ij}}$ is the set of active flows at link $(i,j)$.

The graph shows that in general a decision random variable as a node on the

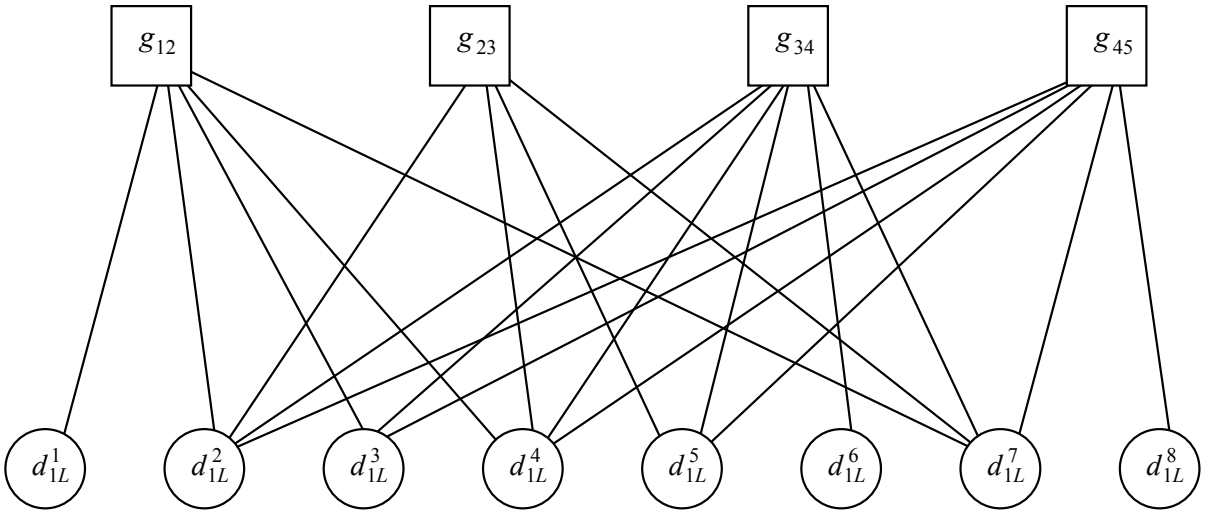



**Figure 22:** Spatial dependence of decision variables

graph can have connections with both near and far neighbors. This shows the global dependence, which results from long-haul flows which extend to far neighbors. This implies that in general, the decision variables are not spatially Markov, and the Gibbs distribution is thus not factorizable.

### 4.3.2 Local Model

If the long-range links can be eliminated in the probabilistic dependency graph, the spatial dependence can be approximated through a spatial Markov model, i.e., a Markov Random Field. Such a Markov Random Field considers only dependence of decision variables with their neighbors, resulting in a truncated Hamiltonian,

$$H^l(\boldsymbol{d}) = \sum_{(i,j),k} \alpha_k B^k d^k_{ij} - \sum_{(i,j),k_1} \sum_{(m,n)\in N_{ij},k_2} \alpha_{k_1} B^{k_1} d^{k_1}_{ij} d^{k_2}_{mn} \delta(k_1,k_2) + \beta \sum_{(i,j)}$$

$$(c_{new} - \sum_k B^k d^k_{ij} - B^0_{ij})^2 U(c_{new} - \sum_k B^k d^k_{ij} - B^0_{ij}),$$

where $N_{ij}$ is an important parameter that contains only neighboring links of $(ij)$. The neighborhood size $N_d$=$|N_{ij}|$.

The corresponding Boltzmann distribution is

$$P^l(\boldsymbol{d}) = Z_0^{-1}\exp\frac{-H^l(\boldsymbol{d})}{T}. \tag{30}$$

$P^l(\boldsymbol{d})$ is an approximated likelihood function,

$$P^l(\boldsymbol{d}) = \prod_{i=1}^{j-1} g_{i,i+1}(\boldsymbol{d}), \tag{31}$$

where $g_{i,i+1}(\boldsymbol{d})$ is a local likelihood function for the connections at link $(i, i+1)$. $g_{i,i+1}(\boldsymbol{d})$ can be further decomposed into all clique potentials associated with connections at link $(i, i+1)$:

$$\begin{aligned} g_{i,i+1}(\boldsymbol{d}) &= \prod_{c\in C_{i,i+1}} \exp(-\psi_c(\boldsymbol{d})) \\ &= \exp(-\sum_{c\in C_{i,i+1}} \psi_c(\boldsymbol{d})), \end{aligned} \tag{32}$$

where $C_{i,i+1}$ is the set of all cliques of link $(i, i+1)$, and $\psi_c(\boldsymbol{d})$ is a potential function of clique $c$.

Consider an example that the neighborhood size $N_d = 2$ for all links, the corresponding factor graph is then the simplest with only nearest neighbor connections, as shown in Figure 23.

## *4.4 Distributed Preemption Algorithms*

We now assume that a local model is a good approximation of the global decision model. The spatial Markov local model then results in a distributed algorithm where nodes can make local decisions on connection preemptions through information exchange with neighbors.

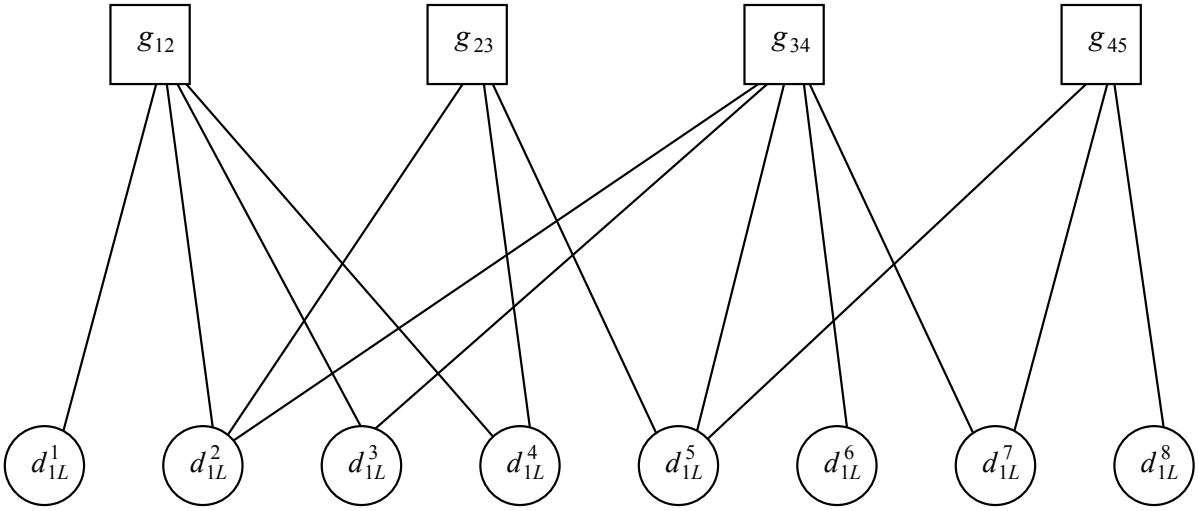



**Figure 23:** Localized spatial dependence of $\boldsymbol{d}$ with Factor Graph

### 4.4.1 Distributed Algorithm

The distributed algorithm obtains a decision that maximizes the approximated likelihood function and minimizes the cost function,

$$\begin{aligned} \hat{\boldsymbol{d}} &= \arg\max_{\boldsymbol{d}} P^l(\boldsymbol{d}) \\ &= \arg\min_{\boldsymbol{d}} H^l(\boldsymbol{d}). \end{aligned} \tag{33}$$

Since $P^l(\boldsymbol{d})$ is factorizable, maximizing the global likelihood function reduces to maximizing the local likelihood function at cliques, i.e., $P^l(\boldsymbol{d_{ij}}|\boldsymbol{d_{N_{ij}}})$ for $1 \leq i \leq L-1$ and $j = i+1$, where $\boldsymbol{d_{N_{ij}}}$ is the set of decision variables of neighboring links. These local likelihoods are functions of the decision variables of neighboring links, and thus the decisions can be updated locally. In addition, the local maximizations result in coupled equations due to the nested Markov dependence, which shows that information exchange is needed only among neighbors.

Maximizing local likelihood functions can be implemented as local learning algorithms at individual nodes. The learning algorithms perform probabilistic inference using either approximated sum product algorithm [55] or stochastic relaxation [33]. The sum product algorithm can be applied to the factor graph in Figure 23. There, the *function nodes* correspond to the local potential function of a link, which consists of active flows that pass the link and the other active flows which are correlated with. The *variable nodes* correspond to the active flows over the preempting route. This algorithm produces an exact solution for a graph that has no loops, and an approximation for a loopy graph [55]. The factor graph of preemption problem is usually loopy, resulting in approximated decisions.

Stochastic relaxation can be applied for each link to make local preemption decisions. Let $\boldsymbol{d}_{\boldsymbol{ij}}^{\boldsymbol{S_F}\backslash\{\boldsymbol{k}\}}$ be a set of decisions on active flows at link $(ij)$, excluding the decision on flow $k$. Let $d_{ij}^k(t+1)$ be an updated decision on flow $f^k$ at the $(t+1)$th iteration and at link $(ij)$. Then,

$$d_{ij}^k(t+1) \;=\; \begin{cases} 1, & \text{with } P^l\left(d_{ij}^k(t+1)=1|\boldsymbol{d}_{\boldsymbol{ij}}^{\boldsymbol{S}\backslash\{\boldsymbol{k}\}}(t), \boldsymbol{d}_{\boldsymbol{N_{ij}}}(\mathrm{t})\right), \\ 0, & \text{with } 1-P^l\left(d_{ij}^k(t+1)=1|\boldsymbol{d}_{\boldsymbol{ij}}^{\boldsymbol{S}\backslash\{\boldsymbol{k}\}}(t), \boldsymbol{d}_{\boldsymbol{N_{ij}}}(t)\right), \end{cases} \tag{34}$$

where

$$P^l\left(d_{ij}^k(t+1)=1|\boldsymbol{d}_{\boldsymbol{ij}}^{\boldsymbol{S}\backslash\{\boldsymbol{k}\}}(t), \boldsymbol{d}_{\boldsymbol{N_{ij}}}(t)\right) \;=\; \frac{\exp\left(-\psi_{ij}(d_{ij}^k(t+1)=1)/T(t+1)\right)}{\sum_{k\in\{-1,1\}}\exp\left(-\psi_{ij}(d_{ij}^k(t+1))/T(t+1)\right)},$$

$$\begin{aligned}\psi_{ij}\left(d_{ij}^k(t+1)=1\right) \;=\; & \alpha_k B^k - \sum_{(m,n)\in N_{i,j},k_1} \alpha_{k_1} B^{k_1} d_{mn}^{k_1}\delta(k,k_1) + \\ & \beta(c_{new} - \sum_{k_1} B^{k_1} d_{ij}^{k_1} - B_{ij}^0)^2 U(c_{new} - \sum_{k_1} B^{k_1} d_{ij}^{k_1} - B_{ij}^0).\end{aligned}$$

A cooling scheduler is applied to the temperature $T(t) = T_0/\log(1+t)$ with $T_0$=3.0. This results in an almost-sure convergence of the algorithm to the global minimum Hamiltonian [33]. That is, with the iterative and distributed updates, the global minimum of the approximated Hamiltonian $H^l(\boldsymbol{d})$ can be reached asymptotically with probability one.

### 4.4.2 Example

We now revisit Figure 21 to show an example of the distributed algorithm. Consider links $(1, 2)$ and $(2, 3)$, and assume that neighborhood size $N_d$=1. That is, a node only exchanges information with its nearest neighbors.

At initial stage, no flows are preempted, i.e. $\{d^1_{12}(0)=0,\ d^3_{12}(0)=0,\ d^1_{23}(0)=0,\ d^4_{23}(0)=0\ \}$. When $t = 1$, the decision variables are updated,

$$d^1_{12}(1) \ = \ \arg\max_{d\in\{0,1\}} P(d^1_{12}(1) = d|d^3_{12}(0), d_{N_{12}}(0)),$$

where $d_{N_{12}}(0) = \{d^1_{23}(0), d^4_{23}(0)\}$. The updated decision $d^1_{12}(1)$ is sent to the neighboring links. This process is applied similarly to the other decision variables. At the second time epoch $(t = 2)$,

$$d^4_{23}(2) \ = \ \arg\max_{d\in\{0,1\}} P(d^4_{23}(2) = d|d^1_{23}(1), d_{N_{23}}(1)),$$

where $d_{N_{23}}(1) = \{\ d^1_{12}(1),\ d^3_{12}(1),\ d^1_{34}(1),\ d^5_{34}(1)\}$.

The process is repeated until an equilibrium state is reached.

### 4.4.3 Information Exchange

The distributed preemption decisions require information exchange with neighbors. The clique structure of the Markov Random Field determines the range of information exchange, which is the neighborhood size $N_d$. The type of the information exchanged is $\boldsymbol{d}_{\boldsymbol{N_{ij}}}(t)$ as in the conditional probability in (34). The amount of information used at a decision making characterizes the communication/computation complexity. The information exchange is per-flow based but moderate when limited to neighbors.

## *4.5 Near-Optimality and Complexity: Analysis*

In this section, we conduct analytical studies to identify a certain sufficient conditions for the near-optimality of the distributed preemption, the communication/computation complexity, and the optimality-complexity trade-offs.

### 4.5.1 Short-range dependent decision variables

When are distributed preemption decisions optimal? To answer this question, we need to consider how well a Markov Random Field approximate the global model. This should be done by studying active flows resulting from traffic patterns since flows are the origin of spatial dependence.

#### *4.5.1.1 Bounded-Length Flows*

Traffic patterns of active flows result in spatial dependence among decision variables. Consider simplified traffic patterns where the hop-count of each active flow is bounded by $h_0$. Then the set of preemption decisions are strictly Markov as shown below.

*Lemma 2: Assume that the hop-count of each active flow is bounded by $h_0$ ($h_0 \geq 1$). Let $N_{mn}^{h_0}$ be the neighborhood of link $(m,n)$. $N_{mn}^{h_0}$ includes all links within $h_0$ hops from $(m,n)$. Let $\boldsymbol{d}_{N_{mn}^{h_0}} = \{\boldsymbol{d_{uv}}$, for $\forall (u,v) \in N_{mn}^{h_0}\}$ denote a set of decisions in the neighborhood, and $\boldsymbol{d} \backslash \boldsymbol{d_{mn}}$ be all decision variables except $\boldsymbol{d_{mn}}$. Then, $P(\boldsymbol{d_{mn}} | \boldsymbol{d} \backslash \boldsymbol{d_{mn}}) = P(\boldsymbol{d_{mn}} | \boldsymbol{d}_{N_{mn}^{h_0}})$.*

The proof is provided in Appendix C. *Lemma 2* shows that the set of decision variables on active flows of a limited span forms a Markov Random Field (MRF), where $h_0$ corresponds to an upper bound of the neighborhood size of the MRF. This is intuitive as flows of a bounded length corresponds to *the disk model* of wireless multi-hop networks.

#### *4.5.1.2 Shortest-Path Flows*

In reality, however, the hop-count of active flows is a variable and cannot be assumed to be bounded. Thus, we study the spatial dependence of decision variables in a more general setting of shortest path flows. Shortest-path flows constitute more realistic

traffic patterns and are thus considered with the following assumptions for analytical convenience:

(1) A network is planar and homogeneous where each node (except edge nodes) has the same nodal degree $d_0$ ($d_0 \geq 2$).

(2) A source-destination pair is chosen randomly from all pairs in the network.

(3) A preempting route is a shortest-path between the source and destination node of a new connection.

(4) Active flows are assumed to take shortest routes from randomly-chosen source-destination pairs whose paths partially coincide with the route of the new connection setup.

We now define a measure of spatial dependence of two links on the preempting route (e.g., $(i-1,i)$ and $(j,j+1)$ for $1 \leq i-1$ and $i \leq j < L$) such as Definition 1.

Definition 1. *Link-Dependency Probability $P_{ij}$*: $P_{ij}$ denotes the probability that a flow uses both links $(i-1,i)$ and $(j,j+1)$, which are separated by $|j-i|$ hops on the preempting route.

The link-dependency probability $P_{ij}$ then characterizes the spatial dependence of any two flows at these two links.

*Lemma 3: Let $P_{ij}^l$ be a lower bound of $P_{ij}$. For shortest-path flows under assumptions (1), (2), (3), and (4), $P_{ij}^l = (\frac{1}{d_0-1})^{|j-i|}$, where $|j-i|$ is the hop distance between links $(i-1,i)$ and $(j,j+1)$.*

The proof is provided in Appendix D. This lemma suggests that the continuity of a shortest-path flow follows at least a geometric probability, where $\frac{1}{d_0-1}$ is the lower

bound of the probability for such a flow to continue at the next hop.

*Lemma 4: Let $P_{ij}^u$ be an upper bound of $P_{ij}$. Consider a network topology of a regular lattice with node degree* 4. *For shortest-path flows under assumptions (1), (2), (3), and (4),*

$$P_{ij} \quad \leq \quad \begin{cases} \frac{C\left(|j-i|, \frac{|j-i|}{2}\right)}{2(2^{|j-i|}-1)}, & |j-i| = 2 \\ \frac{C\left(|j-i|, \frac{|j-i|}{2}\right)}{3(2^{|j-i|}-1)}, & |j-i| > 2, \end{cases} \tag{35}$$

*where $C(a,b)$ is a combinatorial coefficient, $C(a,b)=\frac{a!}{(a-b)!b!}$, and $|j-i| \geq 2$. For $|j-i| >> 1$ , $P_{ij}^u \approx \frac{1}{2\sqrt{2\pi|j-i|}}$ for $|j-i| = 2$; and $\frac{1}{3\sqrt{2\pi|j-i|}}$ for $|j-i| > 2$.*

The proofs can be obtained by counting the number of shortest paths between nodes $i$ and $j$, and is given in Appendix E.

As the hop distance $|j-i|$ increases, both bounds decrease, one exponentially and the other polynomially. Figure 24 depicts both the upper and lower bounds as well as empirical probability $P_{ij}$. The probability $P_{ij}$ is measured on a regular lattice network of 250 nodes, where active flows are routed onto the shortest paths between randomly chosen source-destination pairs. The probability $P_{ij}$ decays fast and its decaying rate is more close to that of the lower bound $P_{ij}^l$ which exponentially decays.

Quantitatively, *Lemma* 3, 4 and the empirical result suggest that on average, shortest-path flows share only a few hops with the preempting route. Thus, Markov Random Fields may be a good approximation to a set of decision variables $\boldsymbol{d}$.

#### *4.5.1.3 General Flows over a Random Topology*

Consider a random topology of $N$ nodes. Node degree of a node $i$ is denoted with $d_0(i)$, which is $1 \leq min(d_0) \leq d_0(i) \leq max(d_0) \leq N$. For two links $(i, i+1)$ and $(j, j+1)$ over a preempting route, let $|j-i|$ denote the hop counts of the shortest-path between node $i$ and $j$ in the random topology. Note that the preempting route may

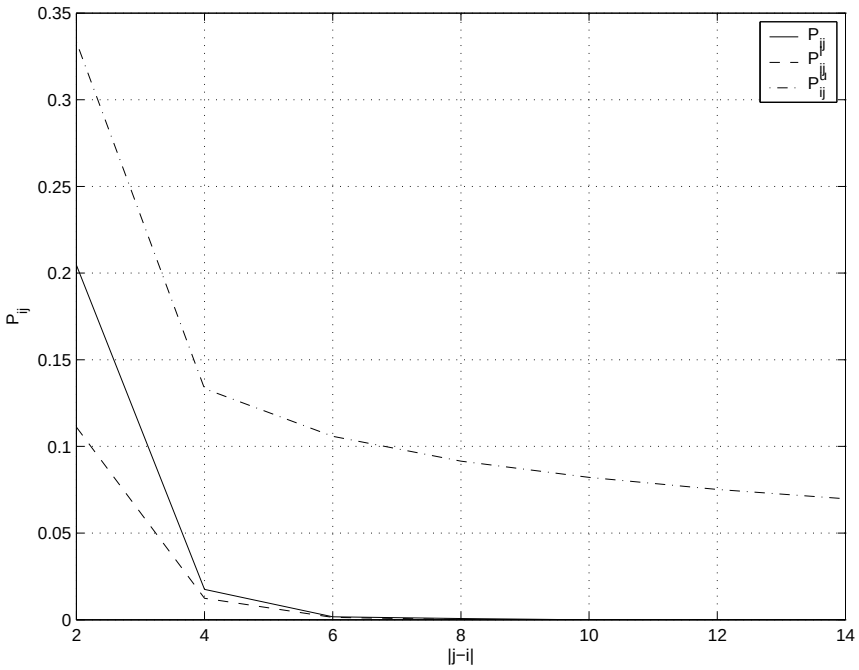



**Figure 24:** Upper and lower bounds of the probability that a flow visits both links $(i-1,i)$ and $(j,j+1)$ on the preempting route

not follow the shortest path.

*Lemma 5:* $P_{ij}^l = (\frac{1}{max(d_0)})^{N-1}$; *and* $P_{ij}^u = (\frac{1}{min(d_0)})^{|j-i|}$.

The proof is obvious and thus omitted. From *Lemma 5*, it is a general phenomenon that the spatial dependence of the preemption decision variables at any two links decays as the distance between two links or the node degree of intermediate nodes increases.

### 4.5.2 Sufficient Conditions for Near-Optimality

We now derive sufficient conditions for near-optimality of distributed preemption.

Definition 2. *Near-optimality of distributed decisions*: Given a set of randomly generated flows in a network, a randomly chosen S-D pair of a new connection is routed

over the network with a proper routing algorithm (e.g., shortest-path first algorithm). Let $\boldsymbol{d^*}$ and $\hat{\boldsymbol{d}}$ be two sets of decisions that minimize the global Hamiltonians $H()$ and its approximation $H^l()$, respectively. The optimality of distributed decisions $\hat{\boldsymbol{d}}$ is measured by the expected value of the difference $\Delta$, where $\Delta = |H(\boldsymbol{d^*}) - H(\hat{\boldsymbol{d}})|$. The difference $\Delta$ is considered to be random when stochastic relaxation is used.

Given a desired performance $\epsilon_{th} > 0$, if $E(\Delta) \leq \epsilon_{th}$, $\hat{\boldsymbol{d}}$ is near-optimal, where $E()$ denotes the expectation with respect to solutions.

We now derive sufficient conditions for the near-optimality. This suffices to investigate whether and when the long-range dependence of active flows can be neglected in the global Hamiltonian. For feasibility of analysis, we consider in this paper traffic patterns with a geometric probability drawn from the previous section.

Let $p_c$ be a continuity probability for a flow to continue to the next link on the preempting route. Such a probability has been used in two other contexts to describe the continuity of a wavelength path in optical networks [7] [65]. $p_c$ characterizes the range of dependence of active flows. In fact, if a flow continues with probability $p_c$ at each link independent of other links, the length of an active flow would obey a geometric probability [65]. For example, $p_c = \frac{1}{d_0 - 1}$ for the lower bound of the continuity probability of the shortest-path flows.

Although a large number of flows are short-range dependent, there can still be long flows. So a sufficient condition for the near-optimality needs to specify when the aggregated effects of long flows are negligible in the truncated Hamiltonian. For feasibility of analysis, we consider a simplified scenario that the variance of bandwidths of active flows is not large.

Theorem 3: *Let $B_0 > 0$ be a constant bandwidth. Let $p_c$ be the probability for a flow to continue at the next hop on preempting route. For given $\epsilon_B$ ($0 < \epsilon_B < 1$),*

*assume that bandwidth $B^k$ of flow $k$ satisfies $|\frac{B^k - B_0}{B_0}| \leq \epsilon_B$ for $\forall\ k$. Then*

$E(\Delta) \leq p_c^{N_d} B_0 \frac{1+\epsilon_B}{1-\epsilon_B} L$,

*where $C$ is link capacity, and $N_d$ is the size of neighborhood.*

The proof is provided in Appendix F. Theorem 3 provides the following observations when active flows follow a geometric distribution.

(a) For a given $p_c$ and $C$, the larger the neighborhood size $N_d$ in the Markov Random Field, the smaller the upper bound, and the better the performance may be for distributed preemption.

(b) The upper bound increases with respect to link capacity $C$ as $\frac{C}{B_0(1-\epsilon_B)}$ characterizes the maximum number of active flows. Thus, the performance of distributed preemption may degrade when link capacity increases.

(c) The upper bound also increases with respect to the route length ($L$).

It should be noted that the above studies of the optimality assumes that the stochastic relaxation is capable of obtaining a global minimum of the global and local models. This holds true as the convergence of the algorithm occurs almost surely [33].

### 4.5.3 Complexity

A key advantage of distributed preemption is the reduced complexity, i.e. the limited information exchange only with neighbors.

Definition 3. *Computation Complexity*: Let $f_{max}$ denote the maximum number of active flows at a link. Let $i_{ter}$ denote the total number of iterations until convergence. Computation complexity (CompC) at a node indicates the complexity of the preemption decisions on the active flows at the node. $CompC = O(f_{max} i_{ter})$.

(34) shows that the complexity for the preemption decision on a flow is $O(1)$, thus the complexity at a node $CompC = O(f_{max} i_{ter})$.

Min-Conn and Min-BW [77] are decentralized algorithms that minimize the number of preempted flows and the amount of preempted bandwidth at each hop, respectively. The complexity of Min-Conn and Min-BW are $O(f_{max}^2)$ and $O(f_{max} 2^{f_{max}})$, respectively.

By managing $i_{ter}$ to be bounded with a proper small value, we can obtain a globally near-optimal decision that is obtained with a smaller complexity than that of decentralized algorithms.

Definition 4. *Communication Complexity*: Let $N_d$ denote the neighborhood size for exchanging information in distributed preemption. Communication complexity (CC) of a node is defined as the total amount of information exchanged at the decision making of the node. $CC = O(N_d f_{max} i_{ter})$.

Hence the communication/computation complexity grows linearly with respect to the neighborhood size and the number of active flows. The communication/computation complexity for a centralized scheme increases linearly with the number of hops on the preempting route $L$. The complexity for distributed preemption, however, is bounded by $N_d$, which is due to local information exchange among neighbors. For the simplest case, $N_d$ can be as small as 1.

### 4.5.4 Optimality and Complexity Trade-off

Reducing the communication/computation complexity results in a simpler local model. The performance, however, may degrade accordingly. Therefore, a trade-off between the optimality and complexity needs to be explored.

Theorem 4: *For a given performance* $\epsilon_{th}$*, if* $N_d \geq log_{p_c}(\frac{\epsilon_{th}}{CL}\frac{1-\epsilon_B}{1+\epsilon_B})$*, then* $E(\Delta) \leq \epsilon_{th}$*.*

Theorem 4 can be obtained directly from Theorem 3 through simple algebraic manipulation, and the proof is provided in Appendix G.

Theorem 4 shows that for a better performance (i.e., smaller error bound $\epsilon_{th}$), the neighborhood size $N_d$ needs to be increased in order of $O(log(\epsilon_{th}))$. Similarly, for a fixed neighborhood size $N_d$, as $N_d$ decreases, the available performance bound gets decreased.

## *4.6 Performance and Complexity: Simulations*

We now study further how the performance of distributed decisions varies with respect to topology (e.g., neighborhood size) and traffic patterns through simulation.

### 4.6.1 Performance Metrics and Simulation Setting

Two performance metrics are used in our simulation study that quantify the effectiveness of distributed preemption in bandwidth savings. One metric is the average preempted bandwidth at a link,

$$\frac{1}{L}\sum_{k} B^k d_{1L}^k, \tag{36}$$

where $B^k$ and $d_{1L}^k$ are the bandwidth and the global preemption decision of flow $k$, respectively. The other metric measures the available bandwidths at individual links, where

$$\frac{1}{L}\left(\sum_{i=1}^{L-1}\sum_{k\in \boldsymbol{f_{i,i+1}}} B^k d_{1L}^k\right). \tag{37}$$

Consider Figure 21 again as an example. Assume the new flow requires bandwidth $c_{new} = 5$, and each active flow occupies the same bandwidths. When three flows 1, 2, and 4 are preempted, the average preempted bandwidth at a link is $(5+5+5)/4 =$

15/4, and the average available bandwidth is $(5+10+5+5)/4 = 25/4$. Note that the average preempted bandwidth at a link can be less than $c_{new} = 5$, and the average available bandwidth is always greater than $c_{new}$.

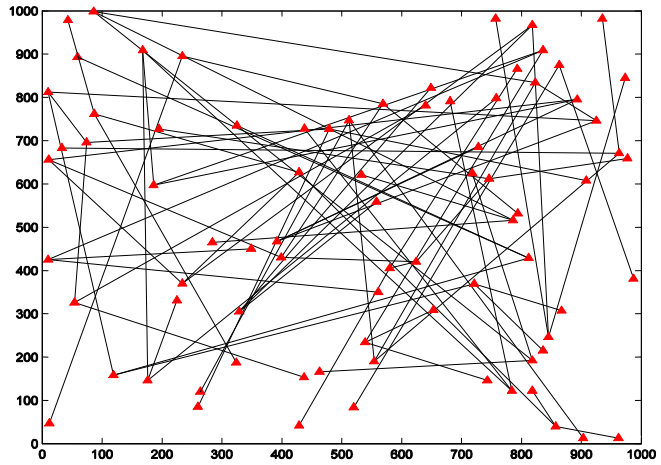
### 4.6.2 Simulation Setting

Our empirical studies consider a multi-class network with two service classes, (i.e., class 1 and 2). For each connection, the source and destination nodes are picked up at random among the nodes in the network. The capacity $C$ of each link is 100 Mbps. The bandwidth of class 1 flows is uniformly distributed between 1.25 and 2.5 Mbps, and that of class 2 flows is $2.5(2k-1)$ Mbps for $1 \leq k \leq 8$. Assume that a new connection belongs to class 2, and thus the required bandwidth $c_{new} = 2.5(2k-1)$ Mbps for $1 \leq k \leq 8$.

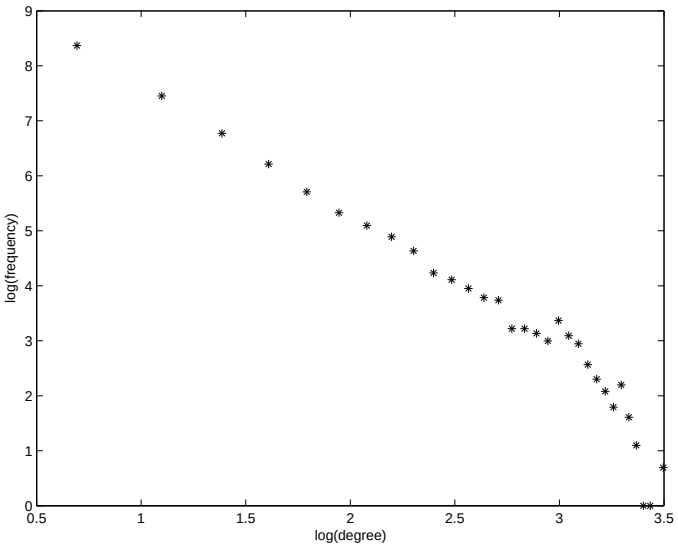
We use both mesh and power-law topologies in the simulations. Figure 25 (a) shows a power law topology with 80 nodes, which is generated through BRITE [18][19]. The histogram of the nodal degree is shown in Figure 25 (b) for a power law topology [19] with 80 nodes. We also have used mesh topologies, e.g. a lattice topology with 100 nodes. For a lattice topology of $d_0 = 4$, the influx link $(i, j)$ onto node $j$ cannot be used for the outflux link. Thus, a lattice topology of $d_0 = 4$ corresponds to the cases of the flow-continuity probability $p_c = \frac{1}{3}$. For a power-law topology, due to clustering, there are only a few hops on the average a S-D pair of each active flow.

The flows of each service class are evenly distributed over the network. There is no traffic at the initial time. The arrival and the departure flows of each class follow a Poisson distribution with arrival rate $(\lambda_i)$ and departure rate $(\mu_i)$, for $i = 1, 2$.

We conduct over 10 experiments with random initial conditions to obtain each curve. The distributed algorithm is used to obtain a set of local decisions. The preemption decision for the flows on the path is then obtained according to Section 4.4. The performance metrics are averaged over all runs.




(a) Power Law Topology with 80 Nodes




(b) Degree Histogram Power Law

**Figure 25:** Topologies used in Performance Evaluation

### 4.6.3 Lattice and Power-Law Topology

For lattice and power-law topologies, multiple active flows are routed over the shortest-path between S-D pairs and fully occupy the network. A new connection setup is done with bandwidth demand $c_{new}$=20 Mbps. The experiments are conducted as described in Section 4.6.2.

Distributed preemption decisions are done by (34) with the change of neighborhood size $0 \leq N_d \leq 2$. Decentralized decisions are done with Min-Conn algorithm

[77] for comparisons. The results are shown in Table 4 and 5. In both topologies, the preemption costs are sharply reduced with the cooperation with neighbors (i.e., $N_d > 0$).

Specifically, in lattice topologies, the node degree $d_0$=4 can be characterized with the flow-continuity probability $p_c$=1/3. The link-dependency probability $P_{ij}$ (in Definition 1) decreases in the order of $O(p_c^{|j-i|})$. Thus, as Table 4 shows, with $N_d = 1$, the preemption cost of distributed decision can be reduced by 50% compared to $N_d = 0$ or comparison algorithm.

In power-law topologies, nodes have heterogeneous degrees, and the path-length of a connection is only a few hops. Thus, the link-dependency probability $P_{ij}$ also decays sharply for $|j-i| >> 1$ (in practice only 2 or 3 hops). The results of power-law topologies are similar to that of lattice topologies.

**Table 4:** Preemption Costs on a Lattice Topology of $d_0$=4

| Comparison | $N_d$=0 | $N_d$=1 | $N_d$=2 |
|---|---|---|---|
| 16.7 | 14.7 | 7.6 | 6.3 |

**Table 5:** Preemption Costs on a Power-Law Topology

| Comparison | $N_d$=0 | $N_d$=1 | $N_d$=2 |
|---|---|---|---|
| 17.2 | 15.9 | 8.8 | 7.5 |

### 4.6.4 Neighborhood Size and Traffic Patterns

In the following subsections, we characterize the topology with the flow-continuity probability $p_c$. At each experiment, active flows are generated randomly for one value of flow-continuity probability $p_c$ and neighborhood size $N_d$. This is repeated for a wide range of parameter $p_c$ and $N_d$ values.

Now consider a preempting route with 10 hops (i.e., $L$=10) and a new connection with bandwidth demand $c_{new}$=20 Mbps. The experiments are conducted as described

in Section 4.6.2.

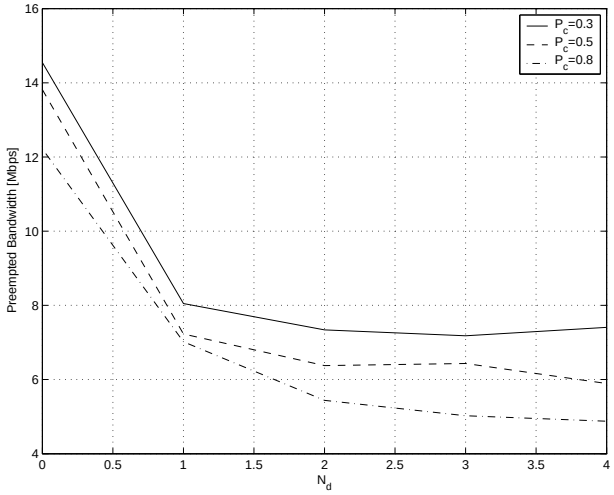
Figure 26 (a) shows that the preempted bandwidth decreases sharply by including the information only from the nearest neighbors. This is especially significant for a small $p_c$ (e.g. $p_c$=0.3) which corresponds to short flows. $N_d$=0 corresponds to decentralized decisions where there is no information exchange with neighbors. Hence the figure shows that the cooperation with the nearest neighbors (i.e., $N_d$=1) can improve the performance by 53%.

The cooperation with far neighbors (e.g., $N_d$=4) produces another 3.3% for $p_c$ = 0.3. But the improvement is not as significant given the increase of computation complexity. Hence, for short flows, the information exchange between the nearest neighbors seems to be sufficient to achieve the near-optimality.

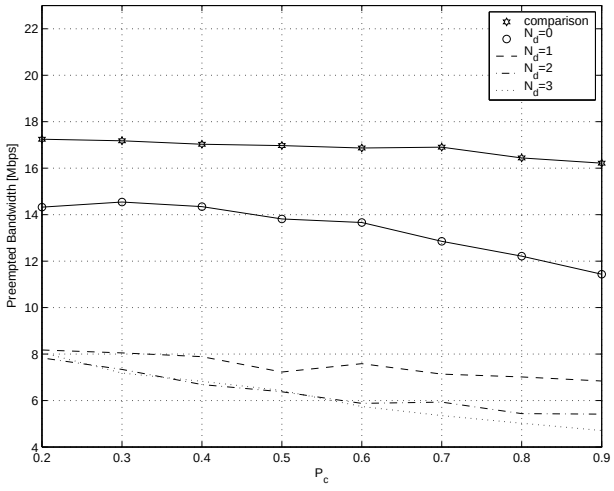
As $p_c$ increases, the dependence among decision variables on different links increases, and the performance gains are more pronounced with a larger neighborhood size. Figure 26 (b) shows that the preempted bandwidth linearly decreases with the increase of $p_c$ for a given $N_d$. As $p_c$ increases, the correlation of any two flows at two different links increases accordingly. Thus, for a given $N_d$, the amount of preempted bandwidth decreases with the increase of $p_c$.

Figure 26 (b) also shows that the preempted bandwidth of the proposed decision algorithm is smaller than that of Min-Conn [77] algorithm. The complexity of Min-BW algorithm is $O(f_{max} \cdot 2^{f_{max}})$, which is computationally intractable. Thus, Min-Conn algorithm (whose complexity is $O(f_{max}^2)$) is used for comparison.

We now examine the average available bandwidth at a link. Note that the available bandwidth that is larger than $c_{new}$ is redundant and undesirable. Figure 27 (a) shows that the redundancy of the available bandwidth decreases sharply by incorporating the information from the nearest neighbors. Again this is pronounced for a small $p_c$ (e.g. $p_c$=0.3). Thus, the cooperation with near neighbors seems to be sufficient for




(a) varying $N_d$



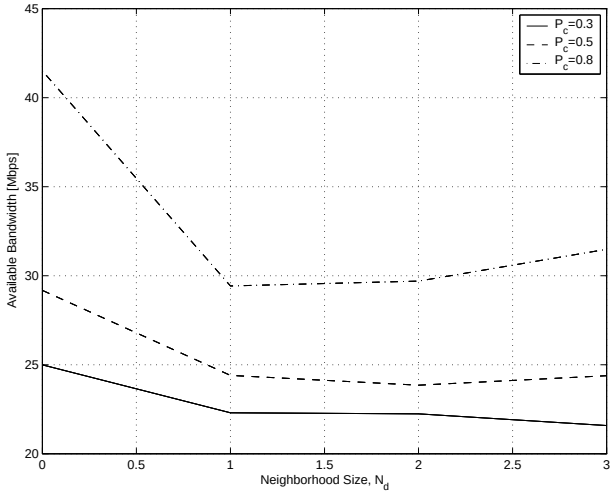

(b) varying $p_c$

**Figure 26:** Average preempted bandwidth, with $c_{new}$=20 Mbps, $C$=100 Mbps, and $L$=10 hops on the preempting route

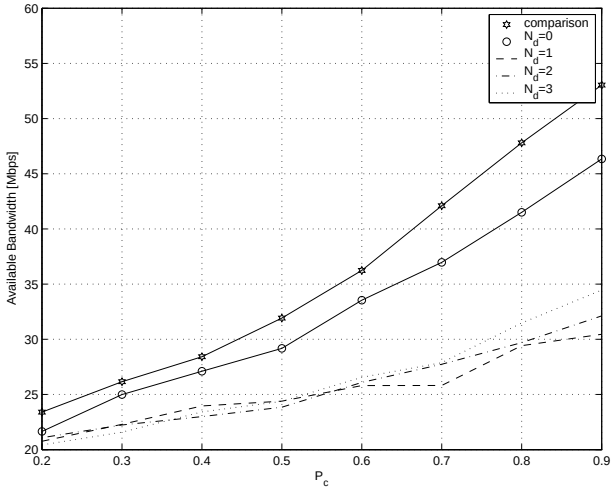
achieving a near-optimal performance with short-range dependent flows.

Figure 27 (b) shows that the available bandwidth linearly increases with the increase of $p_c$ for a given $N_d$. As $p_c$ increases, the correlation between any two connections at two different links increases accordingly. Thus, for a given $N_d$, the available bandwidth increases with the increase of $p_c$.

Figure 27 (b) also shows that the performance of distributed preemption with a larger $N_d$ is closer to the optimal performance where the available bandwidth should be $c_{new} = 20$.




(a) varying $N_d$




(b) varying $p_c$

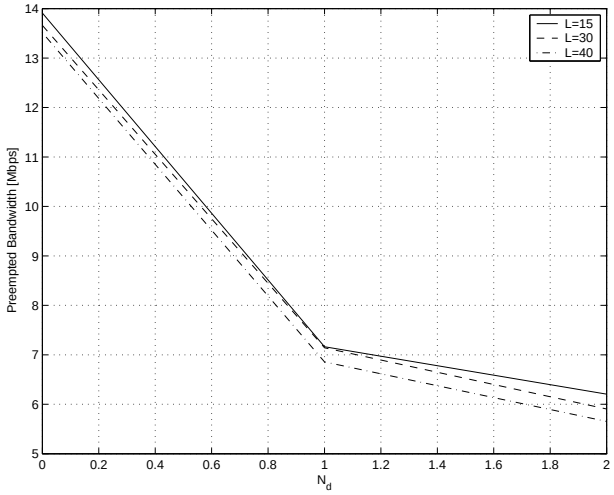
**Figure 27:** Average available bandwidth. $c_{new}$=20 Mbps, $C$=100 Mbps, and $L$=10

### 4.6.5 Path Length

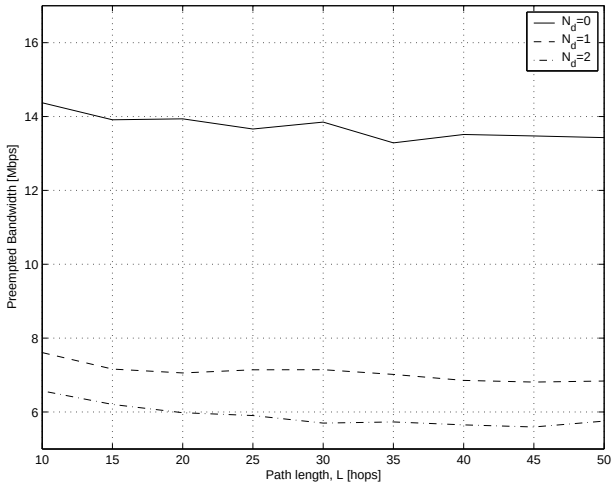
Now we consider the impact of path length $L$ for fixed $p_c$=0.4, $c_{new}$=20 Mbps, and $C$ = 100 Mbps. For a given $N_d$, as $L$ increases, the probability of preempting a flow that is already preempted at another link increases accordingly.

Figure 28 (a) shows that for all $N_d$ values, the corresponding preempted bandwidth decreases as $L$ increases. However, the decrease of preempted bandwidth is lower bounded for $L > 30$ hops, such as Figure 28 (b).

The average available bandwidth shows similar characteristics. Figure 29 (a)




(a) Total preempted bandwidth, varying $N_d$




(b) Total preempted bandwidth, varying $L$

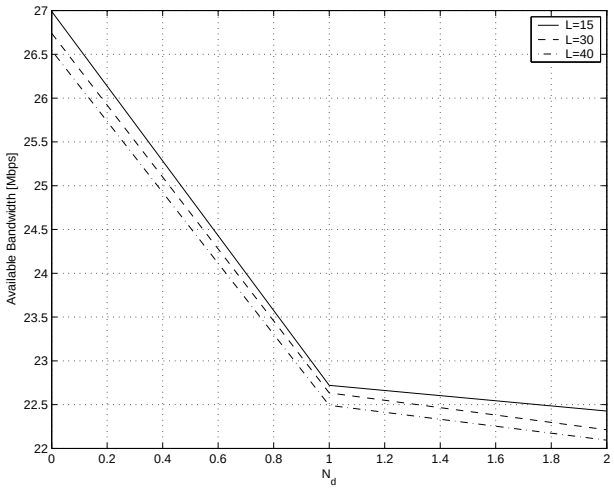
**Figure 28:** Average preempted bandwidth. $c_{new}$=20 Mbps, $C$=100 Mbps, and $p_c$=0.4

shows that for all $N_d$ values, the corresponding available bandwidth decreases with the increase of $L$, which is lower bounded as $L$ increases, such as Figure 29 (b).

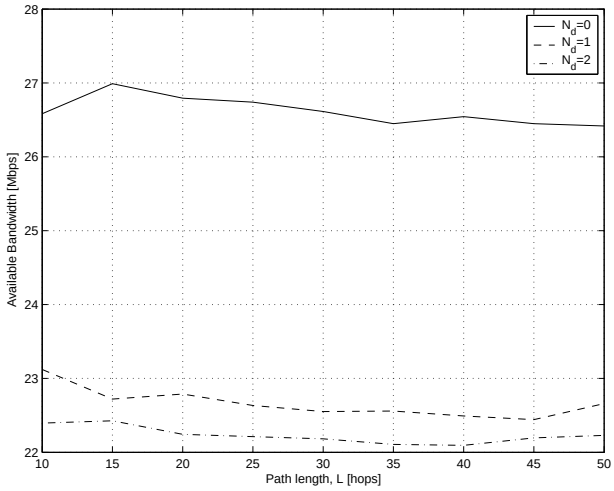
### 4.6.6 Bandwidth Demand

Now we consider the impact of bandwidth demand $c_{new}$ of a new connection together with the neighborhood size. The other parameters are fixed and chosen as $p_c$=0.4, and $C = 100$ Mbps.

Figure 30 shows that for all $N_d$ values, the corresponding preempted bandwidth




(a) Total available bandwidth, varying $L$




(b) Total available bandwidth, varying $L$

**Figure 29:** Average available bandwidth, with $c_{new}$=20 Mbps, $C$=100 Mbps, and $p_c$=0.4

increases as $c_{new}$ increases. Moreover, the gain of cooperative preemption increases also.

## *4.7 Conclusions*

In this work, we have studied distributed connection preemption in multi-class networks. The work is motivated by the fact that connection preemption is known to be NP-complete. Centralized preemption can achieve an optimal performance but

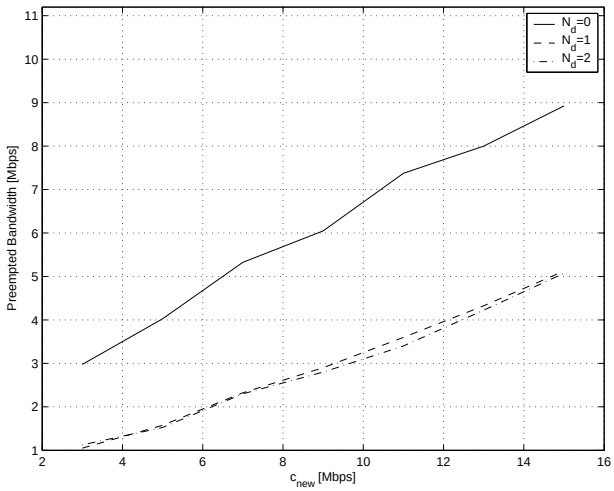



**Figure 30:** Average preempted bandwidth, with $C$=100 Mbps and $p_c$=0.4, varying $c_{new}$

with an intractable communication and computation complexity. Decentralized preemption is computationally feasible but lack of a good performance. This work has focused on whether a near-optimal performance can be achieved by distributed preemption at a moderate communication/computation complexity. We have developed a distributed framework, where nodes make local preemption decisions through cooperation with neighbors. The framework treats distributed preemption as a machine learning problem where a large number of statistically dependent decisions can be treated jointly.

Specifically, we have developed a probabilistic spatial model of distributed preemption decisions. We have shown that a sufficient condition for distributed preemption to be near-optimal is that the spatial model is a Markov Random Field. We have then identified a certain sufficient conditions on when a Markov Random Field holds. In particular, the sufficient conditions examine commonly-used traffic patterns including shortest-path flows. The sufficient conditions then quantify the joint impacts of the flow-continuity probability, the link capacity, the communication/computation complexity of distributed algorithms, and route lengths for short-range dependent active flows.

Based on the probabilistic graphical models, we have applied two distributed algorithms based on stochastic relaxation and message-passing. We have shown through analysis and simulations that for short-range dependent flows, information exchange with only nearest neighbors can significantly improve the performance of preemption. The use of more neighbors would result in a near-optimal performance but the improvement is not as pronounced when the complexity increases.

Future work involves studying computation time in terms of delays and an extension to other traffic patterns including long-range dependent flows.

# CHAPTER V

# CONCLUSION

This chapter summarizes the contribution of this research and provides discussions about the future directions of this research.

## *5.1 Contributions*

The objective of this proposed research is to provide fundamental understandings about the interactions and inter-dependence among cross-layer network parameters, and to find the conditions of the spatial dependence to guarantee the near-optimality of distributed management. Based on the derived probabilistic model, we achieve global management objectives (e.g., link scheduling to maximize the one-hop capacity, preemption-based resource allocation) with a fully distributed management, while providing a certain performance guarantee.

To achieve a fundamental and systemtic modeling, different from top-down approaches in machine learning which usually assume a model in the beginning, we take a bottoms-up approach so that conditions and algorithms on *when* and *how* can be studied through internal network properties and externally imposed management constraints, whose information can be obtained easily. We develop an analytical framework for distributed configuration management of large wireless networks where each node adjusts locally its physical and logical configuration through local information exchange with neighbors. We investigate whether and when a near-optimal global network management can be obtained with local cooperations among nodes.

To get those answers, we first derive a global probabilistic model of a network configuration which characterizes the complex spatial dependence of a set of network variables jointly. The global model is thus determined by these internal network

characteristics and management requirements. We then apply probabilistic graphical models in machine learning to show when the global model can be approximated by local models with a certain neighborhood system. This results in a sufficient condition for distributed configuration management to be nearly-optimal: the global model on a network configuration needs to be approximated within a given accuracy by a local model, which belongs to Markov Random Fields.

The considered network management applications include the fair link-scheduling that maximizes the one-hop capacity, the adaptive configuration management according to random failures and environment changes, and the management robustness upon these random events. For example, for the link scheduling of maximizing the one-hop capacity, we derive a near-optimal distributed algorithm, based on the probabilistic inference algorithm of the probabilistic graphical models.

We believe that this graphical model based approach fits well as the framework of the distributed network management for various networks and applications. That is, the proposed approach can characterize the optimality and the complexity of the distributed managements in a systematic way over different scenarios.

To derive a complexity upper-bound of the policy-based resource allocation (i.e., connection preemption problem), we convert the preemption problem in the domain of a routing problem. The preemption problem can be represented with a flow graph (i.e., a *virtual network topology*), and the least-cost route in the flow graph corresponds to the optimal decision; therefore, the complexity can be derived from that of the routing problem. The complexity is shown to be NP-complete, as known.

To understand when and whether the scalable policy-based resource allocation is feasible in a fully distributed way, we study the cause of complexity of resource allocation. We have shown that the routed paths of active flows cause the spatial dependence on the decisions of different links. To overcome this complexity spatial

dependence, a local dependency model is derived. We have identified sufficient conditions on when the approximation error of the local model resides within an error bound. Moreover, randomized and distributed decision algorithms are proposed over the local model. We have shown that as the cooperating neighborhood size increases, the optimality in terms of preemption costs increases at the cost of the increased complexity in terms of communication and computation. The optimality and complexity are well characterized with the neighborhood system over the neighboring links.

We have shown through analysis and simulations that for short-range dependent flows, information exchange with only nearest neighbors can significantly improve the performance of preemption. The use of more neighbors would result in a near-optimal performance but the improvement is not as pronounced when the complexity increases.

## *5.2 Future Research Directions*

Distributed approach is essential for the scalable management of large networks. There is a remaining open issue and a promising application for the distributed management, which are listed as follows.

Distributed management can be characterized with the randomness, originating from the independent and asynchronous decisions and the insufficient neighborhood size.

We have shown that the ultimate asynchronous decisions of all nodes in the network depend on the sequence of the decision-making of nodes in the network, which causes the local optimal traps. To make the ultimate asynchronous decisions be independent of the random sequence of decision-making of nodes, we have counted on the statistical decisions. To do so, we have proposed to represent the iterative and asynchronous decision-makings on the resource allocation in the wireless networks

with a probabilistic model.

The remaining open issue is about the problem of localized information exchange. That is, because of packet flooding issues during information exchanges, each node can maintain only a partial (or local) logical topology of a large network. We refer to the range of the partial topology information of a node as the interference range of the node. A node cannot consider the residual logical topology outside the interference range on the decisions of scheduled channel-access. In the literature, the residual interference is simply ignored in most cases. However, the residual interference cannot always be ignored; moreover, without a proper consideration of the residual interference, the required global management objectives (e.g., SINR constraint) cannot be guaranteed. We study how to consider the ignored long-range spatial dependence in the distributed decisions.

We show that the iterative, statistical, and asynchronous resource allocations can be represented with a *localized factor graph*, which shows a localized dependence. Asynchronous decision-making with local topology information can be done with message-passing algorithm on the factor graph.

In the cognitive-radio networks, primary connections (i.e., users) can remove ongoing active connections of secondary users. Resource utilization of primary and secondary connections resembles the shared resource management in the multi-class networks with different priorities. We thus propose to formulate the resource allocation problem between the primary and secondary connections in cognitive radio networks with that of the multi-class networks, where there are three priority services (i.e., primary users $>$ secondary users with quality-of-service (QoS) service with bandwidth reservation $>$ secondary users with best-effort service).

The resource allocation can be investigated by considering the spatial dependence

of the decisions at links. We study when the spatial dependence of links can be assumed to be independent with a good approximation. This independence assumption provides a great computational efficiency.

# APPENDIX A

# PROOF OF THEOREM 1

$|\frac{H(\boldsymbol{\sigma}^*|\boldsymbol{X})-H(\hat{\boldsymbol{\sigma}}|\boldsymbol{X})}{H(\boldsymbol{\sigma}^*|\boldsymbol{X})}| \leq \frac{|H(\boldsymbol{\sigma}^*|\boldsymbol{X})-H(\hat{\boldsymbol{\sigma}}|\boldsymbol{X})|^u}{|H(\boldsymbol{\sigma}^*|\boldsymbol{X})|^l}$, where the super-scripts $u$ and $l$ denote an upper and a lower bound of the corresponding quantity.

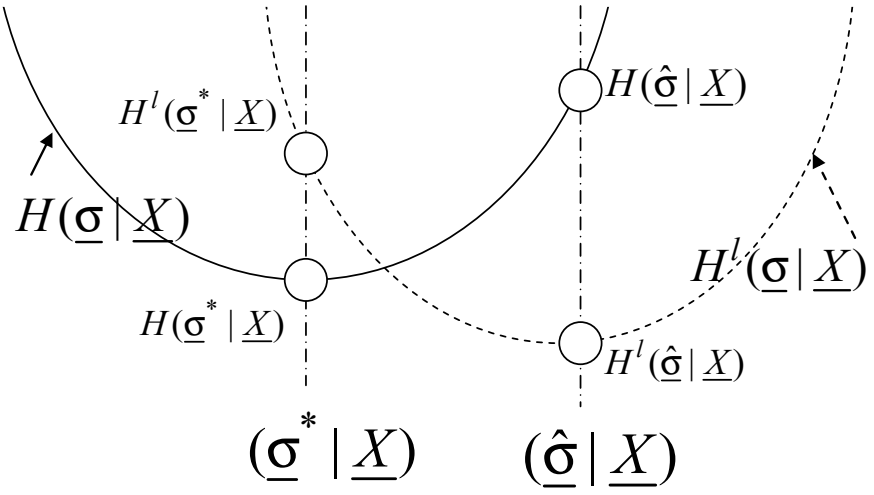



**Figure 31:** Optimal and Sub-optimal Hamiltonians

Figure 31 shows the relative position of related Hamiltonian values in case $H(\boldsymbol{\sigma}^*|\boldsymbol{X})$ $> H(\hat{\boldsymbol{\sigma}}|\boldsymbol{X})$, i.e., $H(\boldsymbol{\sigma}^*|\boldsymbol{X})$, $H(\hat{\boldsymbol{\sigma}}|\boldsymbol{X})$, $H^l(\boldsymbol{\sigma}^*|\boldsymbol{X})$, and $H^l(\hat{\boldsymbol{\sigma}}|\boldsymbol{X})$. Note that $H(\boldsymbol{\sigma}^*|\boldsymbol{X})$ $\leq H(\hat{\boldsymbol{\sigma}}|\boldsymbol{X})$ and $H^l(\boldsymbol{\sigma}^*|\boldsymbol{X}) \leq H^l(\hat{\boldsymbol{\sigma}}|\boldsymbol{X})$. From Figure 31, $|H(\boldsymbol{\sigma}^*|\boldsymbol{X}) - H(\hat{\boldsymbol{\sigma}}|\boldsymbol{X})| \leq$ $|H(\hat{\boldsymbol{\sigma}}|\boldsymbol{X}) - H^l(\hat{\boldsymbol{\sigma}}|\boldsymbol{X})| + |H(\boldsymbol{\sigma}^*|\boldsymbol{X}) - H^l(\boldsymbol{\sigma}^*|\boldsymbol{X})|$. For any configuration $(\boldsymbol{\sigma_a}|\boldsymbol{X_a})$, $|H(\boldsymbol{\sigma_a}|\boldsymbol{X_a})-H^l(\boldsymbol{\sigma_a}|\boldsymbol{X_a})| \leq |R_I(\boldsymbol{\sigma_a}, \boldsymbol{X_a})|+ |R_3(\boldsymbol{\sigma_a}, \boldsymbol{X_a})|$. We denote an upper bound of $|R_3(\boldsymbol{\sigma_a}, \boldsymbol{X_a})|$ and $|R_I(\boldsymbol{\sigma_a}, \boldsymbol{X_a})|$ with $I_3$ and $I_R$, respectively.

$$
\begin{aligned}
\text{(i)}:|R_3(\boldsymbol{\sigma_a}, \boldsymbol{X_a})| &\leq \sum_{k_1=1}^{\frac{r_f}{r_c}} \sum_{k_2=1}^{\frac{r_f}{r_c}} 2P_t k_1^{\frac{-\alpha}{2}} k_2^{\frac{-\alpha}{2}} r_c^{-\alpha} \\
&\leq 2P_t \left(1 + \left(\int_{k=1}^{\frac{r_f}{r_c}} k^{\frac{-\alpha}{2}} dk\right)\right)^2 \cdot r_c^{-\alpha} \\
&= \begin{cases} 2P_t r_c^{-\alpha}(1 + 0.5 \ln \mathcal{C})^2, & \alpha = 2 \\ \\ \frac{2P_t r_c^{-\alpha}}{(\alpha-2)^2}(\alpha - 2(\mathcal{C})^{\frac{2-\alpha}{2}})^2, & \alpha > 2 \end{cases} \\
&= I_3.
\end{aligned}
$$

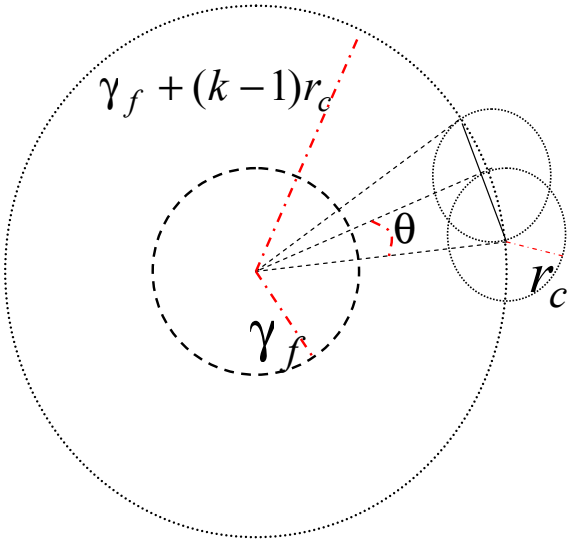



**Figure 32:** Cardinality of the $k$-th Frontier

(ii) $I_R$: For an active dipole $\sigma_{ij}$, the set of neighboring active dipoles at the $k$th frontier is denoted with $G_k$, which is shown in Figure 32. The cardinality of $G_k$ is upper bounded, i.e., $|G_k| \leq \frac{2\pi}{\theta} = \pi / \sin^{-1}\left(\frac{\frac{r_c}{2}}{r_f + (k-1)r_c}\right) < 2\pi(r_f + (k-1)r_c)/r_c$. Refer Figure 32 for the definition of $\theta$.

For an active dipole $\sigma_{ij}$, an upper bound of the residual interference outside interference range is denoted with $I_{k_U}$, i.e.,

$$\begin{aligned} R_{I_{ij}}(\boldsymbol{\sigma}, \boldsymbol{X}) &\leq \sum_{k=1}^{k_U} 2P_t 2\pi \cdot \frac{(r_f + (k-1)r_c)^{\frac{2-\alpha}{2}}}{r_c} (l_{th}(1+\epsilon_0))^{\frac{-\alpha}{2}} \\ &= I_{k_U}, \end{aligned}$$

where $k_U$ is a constant that satisfies quadratic inequalities of

$$\sum_{k=1}^{k_U-1} 2\pi \frac{r_f + (k-1)r_c}{r_c} < \frac{(N-2)l_{th}(1+\epsilon_0)}{r_c} \leq \sum_{k=1}^{k_U} 2\pi \frac{r_f + (k-1)r_c}{r_c}, \tag{38}$$

where the value $\frac{(N-2)l_{th}(1+\epsilon_0)}{r_c}$ denotes an upper bound of the maximum number of available active dipoles except $\sigma_{ij}$ in a network with total $N$ nodes.

From Eq.(38), $k_U$ can be obtained that $k_U^l \leq k_U < k_U^u$, where
$k_U^u = \frac{1}{2}\left((3 - \frac{2r_f}{r_c}) + \sqrt{(\frac{2r_f}{r_c} - 1)^2 + \frac{4}{\pi}(N-2)(\frac{l_{th}(1+\epsilon_0)}{r_c})}\right)$.

In Eq.(38), the summation $\sum_{k=1}^{k_U} g(k)^{-\alpha}$ is upper bounded with the integration of $(g(1)^{-\alpha} + \int_{k=1}^{k_U} g(k)^{-\alpha} dk)$. For our convenience, we first derive an upper bound of the following quantity $(r_f + (k_U^u - 1)r_c)$,

$$\begin{aligned} & r_f + (k_U^u - 1)r_c \qquad (39) \\ \leq\ & r_f + r_c/2(1 - 2r_f/r_c) + r_c/2 \cdot \\ & \sqrt{(\frac{2r_f}{r_c} - 1)^2 + \frac{4}{\pi}\frac{(N-2)l_{th}(1+\epsilon_0)}{r_c}} \\ \leq\ & r_c/2 + r_c/2\left((2r_f/r_c - 1) + 2\sqrt{\frac{Nl_{th}(1+\epsilon_0)}{\pi r_c}}\right) \\ \leq\ & r_f + \sqrt{\frac{Nl_{th}(1+\epsilon_0)r_c}{\pi}} \\ \leq\ & r_f + \sqrt{Nl_{th}r_c}. \end{aligned}$$

By replacing $k_U$ with $k_U^u$ in Eq.(38) and $r_f$ with $\sqrt{\mathcal{C}}r_c$,

$$
\begin{aligned}
& I_{k_U} \\
\leq\ & \frac{2P_t 2\pi}{r_c(l_{th}(1+\epsilon_0))^{\frac{\alpha}{2}}}\left(r_f^{\frac{2-\alpha}{2}} + \int_1^{k_U^u}(r_f+(k-1)r_c)^{\frac{2-\alpha}{2}}dk\right) \\
\leq\ & \begin{cases} \frac{P_t 4\pi}{r_c^2(l_{th}(1+\epsilon_0))^{\frac{\alpha}{2}}}\left(\frac{1}{\sqrt{\mathcal{C}}} + \ln(1+\sqrt{Nl_{th}/(\mathcal{C}r_c)}\right), & \alpha = 4 \\ \frac{P_t 4\pi}{r_c^2(l_{th}(1+\epsilon_0))^{\frac{\alpha}{2}}}\Big(\mathcal{C}^{\frac{2-\alpha}{4}} r_c^{\frac{4-\alpha}{2}} + \frac{2}{4-\alpha}\cdot \\ \quad\left(-(\sqrt{\mathcal{C}}r_c)^{\frac{4-\alpha}{2}} + (\sqrt{\mathcal{C}}r_c + \sqrt{Nl_{th}}r_c)^{\frac{4-\alpha}{2}}\right)\Big), & \alpha \neq 4. \end{cases} \\
=\ & I_R.
\end{aligned}
$$

Thus, $|H(\boldsymbol{\sigma}^*|\boldsymbol{X}) - H(\hat{\boldsymbol{\sigma}}|\boldsymbol{X})| \leq |H(\hat{\boldsymbol{\sigma}}|\boldsymbol{X}) - H^l(\hat{\boldsymbol{\sigma}}|\boldsymbol{X})| + |H(\boldsymbol{\sigma}^*|\boldsymbol{X}) - H^l(\boldsymbol{\sigma}^*|\boldsymbol{X})|$ $\leq 2(I_R + I_3){N_\sigma}^*$, where ${N_\sigma}^*$ is total number of active dipoles in $\boldsymbol{\sigma}^*$ given $\boldsymbol{X}$.

Next, note that $|H(\boldsymbol{\sigma}^*|\boldsymbol{X})| \geq \min\{P_t l_{ij}^{-\alpha} - \sum_{mn\neq ij} P_t l_{mj}^{\frac{-\alpha}{2}}\ l_{ij}^{\frac{-\alpha}{2}}\}\ N_\sigma^*$ for $\forall\ \sigma_{ij} = 1$. For $\sigma_{ij} = 1$, to satisfy a given $\mathrm{SINR}_{th}$, a sufficient condition is $(P_t(\sum_{mn\neq ij} l_{mj}^{\frac{-\alpha}{2}})^2 + N_b)/(P_t(l_{ij}^{\frac{-\alpha}{2}})^2) \leq 1/\mathrm{SINR}_{th}$, thus $(\sum_{mn\neq ij} l_{mj}^{\frac{-\alpha}{2}})^2 \leq (\frac{l_{ij}^{-\alpha}}{\mathrm{SINR}_{th}} - \frac{N_b}{P_t})$. Therefore, $|H(\boldsymbol{\sigma}^*|\boldsymbol{X})|$ $\geq (P_t l_{th}^{-\alpha}(l+\epsilon_0)^{-\alpha} - P_t l_{th}^{\frac{-\alpha}{2}}(l-\epsilon_0)^{\frac{-\alpha}{2}}\sqrt{l_{th}^{\frac{-\alpha}{2}}(1-\epsilon_0)^{\frac{-\alpha}{2}}\mathrm{SINR}_{th}^{-1} - \frac{N_b}{P_t}})\ N_\sigma^*$.

As a result, $|\frac{H(\boldsymbol{\sigma}^*|\boldsymbol{X}) - H(\hat{\boldsymbol{\sigma}}|\boldsymbol{X})}{H(\boldsymbol{\sigma}^*|\boldsymbol{X})}| \leq \epsilon_\Delta$, where $\epsilon_\Delta = 2(I_R + I_3)/I_D$, where $I_D = P_t l_{th}^{-\alpha}(l+\epsilon_0)^{-\alpha} - P_t l_{th}^{\frac{-\alpha}{2}}(l-\epsilon_0)^{\frac{-\alpha}{2}}\sqrt{l_{th}^{\frac{-\alpha}{2}}(1-\epsilon_0)^{\frac{-\alpha}{2}}/\mathrm{SINR}_{th} - \frac{N_b}{P_t}}$.

As a result, in case of $\epsilon_0$=0 (for simple representation),

$$\epsilon_{\Delta} = \begin{cases} \frac{\mathcal{I}}{r_c^2}\left((2+\ln\mathcal{C})^2+\frac{4\pi}{l_{th}}(r_c+\sqrt{Nl_{th}r_c})\right), & \alpha=2 \\ \frac{\mathcal{I}}{r_c^2}\left(\frac{2}{r_c^2}(2-\frac{1}{\mathcal{C}})^2+\frac{4\pi}{l_{th}^2}(\frac{1}{\sqrt{\mathcal{C}}}+\ln(1+\sqrt{\frac{Nl_{th}}{\mathcal{C}r_c}}))\right), & \alpha=4 \\ \frac{\mathcal{I}}{r_c^2}\left(\frac{2(\alpha-2\mathcal{C}^{\frac{2-\alpha}{2}})^2}{(\alpha-2)^2r_c^{\alpha-2}}+\frac{4\pi}{l_{th}^{\frac{\alpha}{2}}}\left(\mathcal{C}^{\frac{2-\alpha}{4}}r_c^{\frac{4-\alpha}{2}}+\frac{2}{4-\alpha}(\right.\right. & \\ \left.\left.-(\sqrt{\mathcal{C}}r_c)^{\frac{4-\alpha}{2}}+(\sqrt{\mathcal{C}}r_c+\sqrt{Nl_{th}r_c})^{\frac{4-\alpha}{2}}\right)\right)\right), & \text{o.w.,} \end{cases}$$

where $\mathcal{I}$= $2/\left(l_{th}^{\frac{-\alpha}{2}}(l_{th}^{\frac{-\alpha}{2}}-\sqrt{l_{th}^{-\alpha}/\mathrm{SINR}_{th}-N_b/P_t})\right)$.

Moreover, $|\frac{H(\boldsymbol{\sigma}^*,\boldsymbol{X}^*)-H(\hat{\boldsymbol{\sigma}},\hat{\boldsymbol{X}})}{H(\boldsymbol{\sigma}^*,\boldsymbol{X}^*)}| \leq \varsigma_\sigma\ |\frac{H(\boldsymbol{\sigma}^*|\boldsymbol{X}^*)-H(\hat{\boldsymbol{\sigma}}|\hat{\boldsymbol{X}})}{H(\boldsymbol{\sigma}^*|\boldsymbol{X}^*)}| + (1-\varsigma_\sigma)\ |\frac{H(\boldsymbol{X}^*)-H(\hat{\boldsymbol{X}})}{H(\boldsymbol{X}^*)}|$. Since the management decisions on ($\boldsymbol{\sigma}$, $\boldsymbol{X}$) are assumed to be done sequentially from $\boldsymbol{X}$ to $\boldsymbol{\sigma}$, $\boldsymbol{X}^*$=$\hat{\boldsymbol{X}}$. Thus, $|\frac{H(\boldsymbol{\sigma}^*,\boldsymbol{X}^*)-H(\hat{\boldsymbol{\sigma}},\hat{\boldsymbol{X}})}{H(\boldsymbol{\sigma}^*,\boldsymbol{X}^*)}| \leq \varsigma_\sigma\ |\frac{H(\boldsymbol{\sigma}^*|\boldsymbol{X}^*)-H(\hat{\boldsymbol{\sigma}}|\boldsymbol{X}^*)}{H(\boldsymbol{\sigma}^*|\boldsymbol{X}^*)}| \leq \varsigma_\sigma\epsilon \leq \epsilon$.

# APPENDIX B

# PROOF OF LEMMA 1

First, note that $hpc_i$ is the highest preempted class only with the information at link $i$ and without considering the information at the other links.

For link $i$, denote the bandwidth of preempted flows, whose priority is greater than $hpc_i$ and lower than $c_{new}$, with $\Delta_i$ (i.e., $\Delta_i = \sum_{k=hpc_i+1}^{c_{new}-1} \Delta_i^k$). Here $\Delta_i^k$ is the amount of preempted bandwidth of class $k$ at link $i$, for $k > hpc_i$.

Note that $\Delta_i$ is due to the preemptions at the other hops on the preempting route. To be specific, at another hop $j$ on the preempting route with $hpc_j > hpc_i$, a flow of class $c$ (for $hpc_i \leq c \leq hpc_j$) that passes through the link $i$ can be preempted and contribute to $\Delta_i$. Refer Figure 20 as an illustration of $\Delta_i$, where $hpc_i$=2.

Due to $\Delta_i$, at link $i$, the effective bandwidth of the new flow can be considered as $b_{new}^{net} = b_{new} - \Delta_i$, which can even change the value $hpc$ of link $i$ (e.g., from class $hpc_i$ to class $k$ with $k < hpc_i$). Therefore, for two links $i$ and $j$ on the preempting route, $hpc_i$ and $hpc_j$ are dependent.

# APPENDIX C

# PROOF OF LEMMA 2

Consider two links $(i, j)$ and $(m, n)$. Let $|(i, j) - (m, n)|$ denote the distance of these two links. The distance of two links is the hop-counts of the shortest path between them.

In case $|(i, j)-(m, n)| \leq h_0$, there may be a flow that shares both links. Otherwise, i.e. $|(i, j) - (m, n)| > h_0$, there cannot be such a flow that shares both links.

Thus, for the active flows of a link $(i, j)$, they can be found only at the links $\{(m, n)\}$ for $|(i, j) - (m, n)| \leq h_0$. That is, for the link $(i, j)$, given the decision information on all active flows at $(m, n)$ for $|(i, j)-(m, n)|$, the link $(i, j)$ comes to have all decision information, done at the other links, about its active flows. Therefore, the decisions at link $(i, j)$ are conditionally independent of the other links outside of $h_0$ hops.

# APPENDIX D

# PROOF OF LEMMA 3

There are multiple routes from node $i$ to node $j$ on the regular topology in assumption *(1)*, one of which is the preempting route. Since the nodal degree of each node is $d_0$, the probability that a connection that passes link $(i-1, i)$ on the preempting route trespasses through the next link $(i, i+1)$ is $(\frac{1}{d_0-1})$. Thus, the probability that an active flow trespasses the links $(i-1, i)$, $\cdots$, $(j, j+1)$ on the preempting route is $(\frac{1}{d_0-1})^{|j-i|}$. Evidently, this corresponds to a lower-bound of $P_{ij}$.

# APPENDIX E

# PROOF OF LEMMA 4

Consider two nodes $i$ and $j$ on the preempting route. Based on the assumption that each connection follows the shortest path between source-destination pair, the upper bound of the probability $P_{ij}$ can be obtained by counting the total number of shortest paths between nodes $i$ and $j$ with $|j-i|$ hops.

An upper bound of the number of shortest paths from node $i$ to $j$ with $|j-i|$ hops is always bounded by $C(|j-i|, \frac{|j-i|}{2})$. To be specific, a shortest-path from node $i$ to $j$ will be composed of $k$ horizontal and $|j-i|-k$ vertical steps, and the total number of shortest paths from node $i$ to $j$ is $C(|j-i|, k)$, for $1 \leq k \leq |j-i|$. And it is obvious that $C(|j-i|, \frac{|j-i|}{2}) = \max\{C(|j-i|, k), \text{ for } 1 \leq k \leq |j-i|\}$. Thus, an upper bound of the number of shortest paths from node $i$ to $j$ with $|j-i|$ hops is $C(|j-i|, \frac{|j-i|}{2})$.

Now, consider a set of nodes which are separated with node $i$ by the distance of $|j-i|$ hops. Here the distance means the hop-counts of a shortest path. We count the total number of shortest paths from node $i$ to the set of nodes, separated by $|j-i|$ hops from node $i$.

Starting from node $i$, we can reach one of such nodes by taking $r$ horizontal steps and $|j-i|-r$ vertical steps, for $1 \leq r \leq |j-i|$, where the direction (e.g. positive or negative) of all vertical steps or all horizontal steps needs to be the same.

For instance, with all positive vertical and horizontal steps, the number of shortest paths with the distance of $|j-i|$ hops from node $i$ is $\sum_{r=1}^{|j-i|} C(|j-i|, r) \cdot C(r, r)$. From binomial formula, $\sum_{r=1}^{|j-i|} C(|j-i|, r) = 2^{|j-i|} - 1$. There are four combinations about the same directions of vertical/horizontal steps. However, the nodes that are located

on the line of radian 0, $\frac{\pi}{2}$, $\pi$, and $\frac{3\pi}{2}$ centered at node $i$ are counted twice. Thus, a lower bound of the total number of shortest-paths from node $i$ to the set of nodes is $2(2^{|j-i|}-1)$ for $|j-i|=2$, and $3(2^{|j-i|}-1)$ for $|j-i|>2$.

Note that an upper bound of probability $P_{ij}$ is the ratio between an upper bound of the total number of shortest paths from node $i$ to $j$ and a lower bound of the total number of shortest paths from node $i$ to the set of nodes, which are separated from node $i$ by $|j-i|$ hops. Thus,

$$P_{ij} \quad \leq \quad \begin{cases} \frac{C\left(|j-i|,\frac{|j-i|}{2}\right)}{2(2^{|j-i|}-1)}, & |j-i|=2 \\ \frac{C\left(|j-i|,\frac{|j-i|}{2}\right)}{3(2^{|j-i|}-1)}, & |j-i|>2. \end{cases} \tag{40}$$

Consider $|j-i| >> 1$. From stirling's approximation, $n! \approx \sqrt{2\pi}\exp(-n)n^{n+0.5}$. Thus, the numerator becomes $\frac{2^{|j-i|+1}}{\sqrt{2\pi|j-i|}}$, and $P_{ij}^u \approx \frac{1}{2\sqrt{2\pi|j-i|}}$.

# APPENDIX F

# PROOF OF THEOREM 3

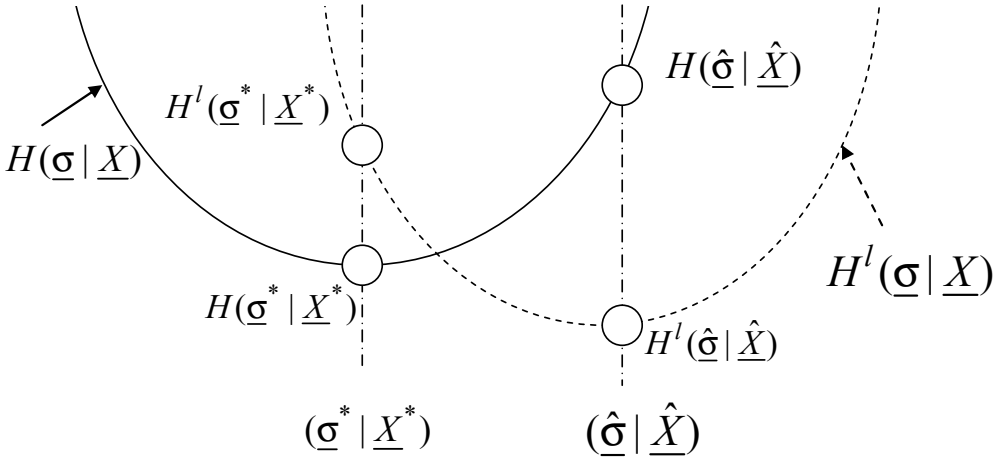



**Figure 33:** Optimal and Suboptimal Hamiltonians

Consider a randomly generated set of active flows in a network and a randomly chosen preempting route that is composed of $L$ hops.

Then, $\Delta = |H(\boldsymbol{d}^*) - H(\hat{\boldsymbol{d}})|$. Figure 33 shows the relative position of related Hamiltonian values $H(\boldsymbol{d}^*)$, $H(\hat{\boldsymbol{d}})$, $H^l(\boldsymbol{d}^*)$, and $H^l(\hat{\boldsymbol{d}})$ for the case of $H(\boldsymbol{d}^*) > H(\hat{\boldsymbol{d}})$. Note that $H(\boldsymbol{d}^*) \leq H(\hat{\boldsymbol{d}})$ and $H^l(\boldsymbol{d}^*) \leq H^l(\hat{\boldsymbol{d}})$. From Figure 33, $|H(\boldsymbol{d}^*) - H(\hat{\boldsymbol{d}})| \leq |H(\hat{\boldsymbol{d}}) - H^l(\hat{\boldsymbol{d}})| + |H(\boldsymbol{d}^*) - H^l(\boldsymbol{d}^*)|$.

Consider an active flow $k$ on a link $(i, i+1)$ of a preempting route. For the local model $H^l(\boldsymbol{d})$ of neighborhood size $N_d$, if the active flow continues on the links $(i+N_d+m,\, i+1+N_d+m)$ or $(i-N_d-m,\, i+1-N_d-m)$ for $m \geq 1$, this continuity cannot be considered with the local model of $N_d$.

From the second term of (28), the error caused by the flow that leaves at link

$(i + N_d + m,\ i + 1 + N_d + m)$ for $m \geq 1$ is $-B^k$ with this continuity probability of $P_c^{N_d+m-1}(1 - P_c)$. Therefore, for an active flow on the preempting route, the total expected error for $\forall m \geq 1$ is less than $B^k \sum_{m=1}^{\infty} P_c^{N_d+m-1}(1 - P_c) = B^k P_c^{N_d}$.

Thus, for any configuration $\boldsymbol{d}$, from (28) and (30), the total expected error caused by all active flows on a preempting route is $E(|H(\boldsymbol{d}) - H^l(\boldsymbol{d})|)$, which is equal to $E(\Delta)$.

$$E(\Delta) \leq \boldsymbol{f_{max}} p_c^{N_d} B_0(1 + \epsilon_B)L \leq p_c^{N_d} B_0 \frac{1 + \epsilon_B}{1 - \epsilon_B} L, \tag{41}$$

where $\boldsymbol{f_{max}}$ denotes the maximum number of active flows at a link. $\boldsymbol{f_{max}} = \frac{C}{B_0(1-\epsilon_B)}$, where $C$ is the link capacity.

# APPENDIX G

# PROOF OF THEOREM 4

From Theorem 3, $E(\Delta) \leq p_c^{N_d} C \frac{1+\epsilon_B}{1-\epsilon_B} L$. If the right term of the above inequality is less than or equal to $\epsilon_{th}$, $E(\Delta) \leq \epsilon_{th}$.

Thus, from simple algebraic manipulations, if $p_c^{N_d} \leq \frac{\epsilon_{th}}{CL} \frac{1-\epsilon_B}{1+\epsilon_B}$, then $\Delta \leq \epsilon_{th}$.

Apply a log function to both sides of the above inequality, for $0 < p_c < 1$. Since $\log(p_c) < 0$ for $0 < p_c < 1$, if $N_d \geq \log_{p_c}(\frac{\epsilon_{th}}{CL} \frac{1-\epsilon_B}{1+\epsilon_B})$, then $\Delta \leq \epsilon_{th}$ and also $E(\Delta) \leq \epsilon_{th}$.

# REFERENCES


[1] A. Agarwal, and J. W. Atwood, *"A Unified Approach of Fault-Tolerance in Communication Protocols Based on Recovery Procedures, " IEEE/ACM Trans. On Networking*, 1996.

[2] T. Anjali, C. Scoglio, and I. Akyildiz, *"LSP and .SP Setup in GMPLS Networks,"* In *Proc. of IEEE Infocom*, 2004.

[3] J. Ash, *"Max Allocation with Reservation Bandwidth Constraints Model for Diffserv-aware MPLS Traffic Engineering and Performance Comparisons," IETF RFC 4126*, June 2005.

[4] A. Akella, G. Judd, S. Seshan, and P. Steenkiste, *"Self-Management in Chaotic Wireless Deployments,"* Proc. *ACM Mobicom*, 2005.

[5] W. R. Ashby, *"Principles of the Self-Organizing Dynamic System, " J. Gen. Psych.*, vol. 37, pp. 125-128, 1947.

[6] B. Barak, S. Halevi, A. Herzberg, and D. Naor, *"Clock Synchronization with Faults and Recoveries, "* Proc. *ACM symposium on Principles of Distributed Computing*, 2000.

[7] R. A. Barry and P. A. Humblet, *" All-Optical Networks with and Without Wavelength Changers," IEEE Journal on Selected Areas in Communications* , vol. 14, pp. 858-867, June 1996.

[8] F. Baker, and J. Polk, *"Implementing MLPP for Voice and Video in the Internet Protocol Suite," Internet draft: draft-baker-tsvwg-mlpp-that-works-02.txt*, October 2004.

[9] A. Banerjea, *"Fault Recovery for Guaranteed Performance Communications Connections," IEEE/ACM Trans. Networking,* vol. 7, pp.653-668, 1999.

[10] J. Baras, and X. Tan, *"Control of Autonomous Swarms Using Gibbs Sampling,"* Proc. *IEEE CDC*, Dec. 2004.

[11] C. Beard and V. Frost, *"Prioritized Resource Allocation for Stressed Networks," IEEE/ACM Transactions on Networking*, vol. 9, no. 5, pp. 618-633, OCTOBER 2001.

[12] H. Xiao and C. Beard, *"Support for High Priority Traffic Using Preemption," http://unofficial.umkc.edu/beardc/BeardXiao_Preempt.pdf*, UMKC SICE Technical Report.

[13] D. P. Bertsekas, R. G. Gallager, *" Data Networks," Prentice Hall*, December 1991.

[14] K. Birman, R. v. Renesse, and W. Vogels, *"Navigating in the Storm: Using Astrolabe for Distributed Self-Configuration, Monitoring and Adaptation,"* Proc. *Active Middleware Workshop (AMS)*, June 2003.

[15] P. Bjorklund, P. Varbrand, and D. Yuan, *"Resource Optimization of Spatial TDMA in Ad Hoc Radio Networks: A Column Generation Approach,"* Proc. *IEEE Infocom*, April 2003.

[16] F. Blanchy, L. Melon, and G. Leduc, *" Routing in a MPLS network featuring preemption mechanisms,"* In Proc. *10th International Conference on telecommunications*, 2003.

[17] G. Brar, D. M. Blough, and P. Santi, *"Computationally Efficient Scheduling with the Physical Interference Model for Throughput Improvement in Wireless Mesh Networks,"* In Proc. *ACM Mobicom*, 2006.

[18] Brite, *http://www.cs.bu.edu/brite/*

[19] T. Bu, and D. Towsley, *"On Distinguishing between Internet Power Law Topology Generators,"* In Proc. *IEEE Infocom*, vol. 2, pp. 638-647, June 2002.

[20] N. Bulusu, J. Heidemann, and D. Estrin, *"GPS-less Low Cost Outdoor Localization for very small devices," IEEE Personal Communications Magazine*, vol. 7, pp. 28-34, Oct. 2000.

[21] Z. Butler, K. Kotay, D. Rus, and K. Tomita *"Generic Decentralized Control for a Class of Self-Reconfigurable Robots,"* Proc. *ICRA*, May 2002.

[22] G.L. Choudhury, L.K. Leung, and W. Whitt, *"An Algorithm to compute blocking probabilities in multi-rate multi-resources loss models," Adv. App-Prob.*, vol. 27, pp. 1104-1143, 1995.

[23] G.L. Choudhury, L.K. Leung, and W. Whitt, *"Efficiently providing multiple grades of services with protection against overloads in shared resources," AT&T Tech. J.*, vol. 74, pp. 50-63, 1995.

[24] M. Chiang, *"Distributed Network Control Through Sum Product Algorithm on Graphs,"* Proc. *IEEE Globecom*, Nov. 2002.

[25] P. Demeester, M. Gryseels, A. Autenrieth, et al., *"Resilience in multilayer networks," IEEE Comm. Magazine*, Aug. 1999.

[26] A. Contessa, *"An Approach to Fault Tolerance and Error Recovery in a Parallel Graph Reduction Machine; MaRs- a Case Study, " ACM SigArch Computer Architecture News*, 1988.

[27] S. Du, A. Khan, S. PalChaudhuri, A. Post, A. K. Saha, P. Druschel, D. B. Johnson, and R. Riedi, *"Safari: A Self-Organizing Hierarchical Architecture for Scalable Ad Hoc Networking Networking," Computer Science TR04-433*, 2004.

[28] T. Elbatt, and A. Ephremides, *"Joint Scheduling and Power Control for Wireless Ad Hoc Networks," IEEE Trans. Wireless Communications*, vol. 3, pp.74-85, Jan. 2004.

[29] D. Estrin, R. Govindan, J. Heidemann, and S. Kumar, *"Next Century Challenges: Scalable Coordination in Sensor Networks,"* Proc. *ACM Mobicom*, Aug. 1999.

[30] G. J. Foschini, Z. Miljani, *"A Simple Distributed Autonomous Power Control Algorithm and its Convergence, " IEEE Trans. Vehicular Technology*, vol. 42, pp.641-647, 1993.

[31] S. Gandham, M. Dawande, and R. Prakash *"Link Scheduling in Sensor Networks: Distributed Edge Coloring Revisited,"* Proc. *IEEE Infocom*, Mar. 2005.

[32] J.A. Garay and I.S. Gopal, *"Call Preemption in Communication Networks,"* Proc. *IEEE Infocom*, 1992.

[33] S. Geman, and D. Geman, *"Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images," IEEE Trans. PAMI*, vol.6, pp.721-741, June 1984.

[34] S. Ghosh, and I.W. Marshall, *" Simple Model of Learning and Collective Decision Making during Nectar Source Selection by Honey Bees," European Conference on Artificial Life (ECAL)*, September 2005.

[35] M. G. Gouda, and N. J. Muhari, *"Stabilizing communication protocols, " IEEE Trans. Computers*, vol. 40, pp-448-458, 1991.

[36] A. Goldsmith, and S.B. Wicker, .Design challenges for energy-constrained ad hoc wireless networks,. IEEE Wireless Communications Magazine vol. 9, pp.8-27, 2002.

[37] M. Greiner, *" Self-organizing control of network structure in wireless communication,"* Proc. *IMA Wireless Communications Workshop*, June 2005.

[38] P. Gupta, and P.R. Kumar, *"The Capacity of Wireless Networks," IEEE Trans. Information Theory*, vol. 46, no. 2, pp. 388-404, Mar. 2000.

[39] S. Herzog, *"Signaled Preemption Priority Policy Element," IETF RFC 2751*, January 2000.

[40] T. Holliday, N. Bambos, P. Glynn, and A. Goldsmith, *"Distributed Power Control for Time Varying Wireless Networks: Optimality and Convergence," Allerton*, 2003.

[41] K. Huang, *"Statistical Mechanics," John Wiley & Sons.*

[42] Z. Huang, and C. Shen, *"Distributed Topology Control Mechanism for Mobile Ad Hoc Networks with Swarm Intelligence, "* Proc. *ACM MobiHoc*, 2003.

[43] J. Huaux, T. Gross, J. Boudec, and M. Vetterli, *"Toward Self-Organized Mobile Ad Hoc Networks: The Terminodes Project," IEEE Network*, pp. 118-124, January 2001.

[44] A. Jadbabaie, J. Lin, and A. S. Morse, *"Coordination of Groups of Mobile Autonomous Agents Using Nearest Neighbor Rules, " IEEE Trans. Automatic Control,* vol. 48, pp.988-1001, 2003.

[45] J.C. de Oliveira, C. Scoglio, I.F. Akyildiz, and G. Uhl, *"A New Preemption Policy for DiffServ-Aware Traffic Engineering to Minimize Rerouting,"* In *Proc. of IEEE Infocom*, vol. 2, pp. 695-704, June 2002.

[46] K. Jain, J. Padhye, V. Padmanabhan, L. Qiu, *"Impact of Interfernece on Multihop Wireless Network Performance,"* Proc. *ACM Mobicom*, 2003.

[47] S. Jeon, R. Abler, J. Copeland, and Y. Pan, *"Path Selection with Class Distribution Information in the Integrated Network,"* *IEEE Communications Letters*, vol. 6, no. 2, pp. 88-90, 2002.

[48] S. Jeon and R.T. Abler, *"Formulation and Optimization of the Connection Preemption Problem,"* *Journal of Computer Communications, Elsevier* vol. 27/3 pp. 253-261, 2004.

[49] S. Jeon, R.T. Abler, and A.E. Goulart, *"The Optimal Connection Preemption Algorithm in a Multi-class Network,"* In Proc. *IEEE ICC*, vol. 4, pp. 2294-2298, April 2002.

[50] S. Jeon, *"Topology Aggregation: Merged-Star Method for Multiple Non-Isomorphic Topology Subgraphs, "* *Journal of Computer Communications, Elsevier* Vol. 29, Issue 11, PP. 1959-1962, July 2006.

[51] S. Jeon and C. Ji, *"Role of machine learning in configuration management of ad hoc wireless networks,"* In Proc. *ACM SIGCOMM, MineNet Workshop*, pp. 223-224, August 2005.

[52] S. Jeon and C. Ji, *"Nearly Optimal Distributed Configuration Management Using Probabilistic Graphical Models,"* Proc. *IEEE MASS, RPMSN workshop*, pp. 219-226, Nov. 2005.

[53] M. Jordan, and Y. Weiss, *"Graphical Models: Probabilistic Inference," Handbook of Neural Networks and Brain Theory*, 2002.

[54] M. Junginger, and Y. Lee, *"A Self-Organizing Publish/Subscribe Middleware for Dynamic Peer-to-Peer Networks," IEEE Network*, pp. 38-59, January/February 2004.

[55] F. Kschischang, B. Frey, H. Loeliger *"Factor graphs and the sum-product algorithm," IEEE Trans. on Information Theory,* vol. 47, no. 2, pp. 498-519, Feb. 2001.

[56] L. Kant, A. McAulery, R. Morera, *"Fault Localization and Self-healing with Dynamic Domain Configuration, "* Proc. *IEEE Milcom*, 2003

[57] Y. Kakuda, T. Kikuno, *"Verification of Responsiveness for Communication Protocols, " IEICE Jpn.,* 1991.

[58] B. Ko, and D. Rubenstein, *"A Distributed, Self-stabilizing Protocol for Placement of Replicated Resources in Emerging Networks,"* Proc. *ACM ICNP*, Nov. 2003.

[59] J. F. Kurose, and K.W. Ross, *"Computer Networking: A Top-Down Approach Featuring the Internet," Addison-Wesley*, 2004.

[60] P. Kyasanur, and N. Vaidya *"Detection and Handling of MAC Layer Misbehavior in Wireless Networks,"* Proc. *DSN*, June 2003.

[61] H. Levine, and W. Rappel, *"Self-organization in Systems of Self-propelled Particles," Physical Review*, vol. 63, 2001.

[62] J. Li, C. Blake, D. De Couto, H. Lee, and R. Morris, *"Capacity of Ad Hoc Wireless Networks,"*, Proc. *ACM Mobicom*, July 2001.

[63] S. Z. Li, *"Markov Random Field Modeling in Computer Vision," Springer-Verlag.*

[64] G. Liu, and C. Ji, *"Cross-Layer Graphical Models for Resilience of All-Optical Networks under Crosstalk Attacks," under submission.*

[65] G. Liu, C. Ji, and V. Chan, *"On the Scalability of Network Management Information for Inter-Domain Light Path Assessment,", IEEE/ACM Trans. Networking*, vol. 13, no. 1, pp. 160-172, March 2005.

[66] H. Luo, S. Lu, V. Bharghavan, J. Cheng, and G. Zhong, *"A Packet Scheduling Approach to QoS Support in Multihop Wireless Networks," ACM MONET*, vol. 9, no. 3, pp. 193-206, June 2004.

[67] Q. Ma and P. Steenkiste, *"Supporting Dynamic Inter-Class Resource Sharing: A Multi-Class QoS Routing,"* In *Proc. of IEEE Infocom*, vol. 2, pp. 649-660, March 1999.

[68] R. Madan, S. Cui, S. Lall, and A. Goldsmith, *"Cross-Layer Design for Lifetime Maximization in Interference-Limited Wireless Sensor Networks" IEEE Trans. Wireless Communications,* vol. 5, No. 11, pp. 3142-3152, Nov. 2006.

[69] K. Manousakis, J. Baras, A. McAuley, and R. Morera *" Network and Domain Autoconfiguration: A Unified Approach for Large Dynamic Networks," IEEE Comm. Magazine*, vol. 43, no.8, pp. 78-85, Aug. 2005.

[70] S. Makam, V. Sharma, K. Owens, and C. Huang, *"Protection/Restoration of MPLS Networks," IETF DRAFT* `draft-makam-mpls-protection-00.txt`, April 2000.

[71] International Telecommunication Union, *"Multi-level Precedence and Preemption Service (MLPP)," Recommendation I.255.3*, 1990.

[72] A. McDonald, and T. Znati, *"A Mobility Based Framework for Adaptive Clustering in Wireless Ad-Hoc Networks," IEEE Journal of Sel. Area of Comm.*, 1999.

[73] J. Modestino, and J. Zhang, *"A Markov Random Field Model-Based Approach to Image Interpretation," IEEE Trans. PAMI*, June 1992.

[74] M. R. Meyer, J. Vasseur, D. Maddux, C. Villamizar, A. Birjandi, *" MPLS Traffic Engineering Soft Preemption," Internet draft: draft-ietf-mpls-soft-preemption-08.txt* , October 2006.

[75] J. Oliveira, C. Scoglio, I. Akyildiz, and G. Uhl, *"New Preemption Policies for DiffServ-Aware Traffic Engineering to Minimize Rerouting in MPLS Networks," IEEE/ACM Trans. on Networking*, vol. 12, no .4, pp. 733-746, August 2004.

[76] S. Park, K. Kim, D. Kim, S. Choi, and S. Hong, *"Collaborative QoS Architecture between DiffServ and 802.11e Wireless LAN,"* In Proc. *IEEE VTC*, vol. 2, pp. 945-949, April 2004.

[77] M. Peyravian and A.D. Kshemkalyani, *"Connection Preemption : Issues, Algorithms, and a Simulation Study,"* In Proc. *IEEE Infocom*, vol. 30, pp. 1029-1043, April 1997.

[78] J. Pearl, *" Probabilistic Reasoning in Intelligent systems: Networks of Plausible Inference," Morgan Kaufmann,* 1988.

[79] G. Pottie, and W. Kaiser, *"Wireless integrated network sensors," Communications of the ACM*, vol. 43. no. 5, May 2000.

[80] C. Prehofer, and C. Bettstetter, *"Self-organization in Communication Networks: Principles and Design Paradigms, " IEEE Communications Magazine*, pp. 78-85, July 2005.

[81] T. Rappaport, *"Wireless Communications: Principles and Practice," Prentice Hall.*

[82] T. Robertazzi and P. Sarachik, *"Self-Organizing Communication Networks, " IEEE Comm. Mag.*, vol. 24, pp. 28-33, 1986.

[83] V. Rodoplu, and T. Meng, *"Minimum Energy Mobile Wireless Networks,"*, *IEEE J. Sel. Areas in Comm.*, vol. 17, no. 8, pp. 1333-1344, Aug. 1999.

[84] L. B. Ruiz, T. R. Braga, F. A. Silva, H. Assuncaso, J. Marcos, S. Nogueira, A. Loureiro, *"On the Design of a Self-Managed Wireless Sensor Networks, " IEEE Comm. Magazine*, 2005.

[85] S. Shakkottai, R. Srikant, and N. Shroff, *"Unreliable Sensor Grids: Coverage, Connectivity and Diameter," IEEE Infocom*, 2003.

[86] S. Shew, *"Fast Restoration of MPLS Label Switched Paths," IETF DRAFT* `draft-shew-lsp-restoration-00.txt`, Oct. 1999.

[87] P. Smyth, D.Heckerman, and M. Jordan, *"Probabilistic independence networks for hidden Markov probability Models," Neural computation*, vol. 9, no. 2, pp.227-269, 1997.

[88] W. Spears, D. Spears, J. Hamann, and R. Heil, *"Distributed, Physics-Based Control of Swarms of Vehicles," Kluwer Autonomous Robots*, vol. 17, pp. 137-162, Aug. 2004.

[89] S. Staab, et.al, *"Neurons, Viscose Fluids, Freshwater Polyp Hydra-and Self-Organizing Information Systems, " IEEE Intell. Sys.*, vol. 18, pp. 72-86, 2003.

[90] V. Stanisic, and M. Devetsikiotis, *"A Dynamic Study of Providing Quality of Service Using Preemption Policies with Random Selection*, In Proc. *IEEE ICC*, vol. 3, pp. 1543-1546, May 2003.

[91] M. Steenstrup, *"Emergent Structure Among Self-Organizing Devices,"* Proc. *NSF-RPI Workshop on PCN*, April 2004.

[92] M. Steinder and A.S. Sethi, *"Probabilistic Fault Localization in Communication Systems Using Belief Networks, " IEEE/ACM Trans. On Networking*, 2004

[93] B. Szviatovszki, . Szentesi, and A. Juttner, *" Minimizing re-routing in MPLS networks with preemption-constraint-based routing," Computer Communications Journal, Elsevier*, vol. 25, pp. 1076-1084, 2003.

[94] T. Tsai, and C. Tu, *"An Adaptive IEEE 802.11 MAC in Multihop Wireless Ad Hoc Networks Considering Large Interference Range,"* Proc. *IFIP WONS*, Jan. 2004.

[95] J. Tsitsiklis and M. Athens, *" On the Complexity of Decentralized Decision Making and Detection Problem," IEEE Trans. Automatic Control* , vol. 30, no. 5, pp.440-446, May 1985.

[96] S. Xu, and T. Saadawi, *"Does the IEEE 802.11 MAC Protocol Work Well in Multihop Wireless Ad Hoc Networks?," IEEE Comm. Magazine*, vol. 39, no. 6, pp.130-137, June 2001.

[97] K. Xu, M. Gerla, and S. Bae, *"How effective is the IEEE 802.11 RTS/CTS Handshake in Ad Hoc Networks?,"* Proc. *IEEE Globecom*, Nov. 2002.

[98] R. J. Vanderbei, *" Linear Programming: Foundations and Extensions," Springer*, 2001.

[99] M. Vieira et al., *"Scheduling nodes in wireless sensor networks: A Voronoi Approach, "* Proc. *IEEE LCN*, pp. 423-429, Oct. 2003.

[100] R. C. Vieira and P. R. Guardieiro, *"A proposal and evaluation of a LSP preemption policy implemented with fuzzy logic and genetic algorithms in a Diff-Serv/MPLS test-bed,"* In Proc. *International Conference on Communications, Circuits and Systems*, May 2005.

[101] A. Wagner, and V. Anantharam, *" Wireless sensor network design via interacting particles,"* Proc. *Allerton*, Oct. 2002.

[102] G. Wang, G. Cao, T.L. Porta, W. Zhang, *"Sensor relocation in mobile sensor networks,"* Proc. *Infocom*, Mar. 2005.

[103] R. Wattenhofer, L. Li, P. Bahl, and Y. Wang, *"Distributed topology control for power efficient operation in multihop wireless networks,"* Proc. *IEEE Infocom*, April 2001.

[104] E.W.M. Wong, A.K.M. Chan, and T.P. Yum, *"Analysis of Rerouting in Circuit-Switched Networks," IEEE/ACM Transactions on Networking*, Vol. 8, No. 3, pp. 419-427, June 2000.

[105] K. Yu, L. Zhang, and H. Zhang, *" A Preemption-aware Path Selection Algorithm for DiffServ/MPLS Networks," IEEE Workshop on IP Operations and Management*, 2004.

[106] S. Zachary, and I. Ziedins, *"Loss Networks and Markov Random Fields," Journal of Applied Probability*, vol. 36, no.2, pp.403-414, 1999.

[107] H. Zhang, and A. Arora, *"GS3: Scalable Self-Configuration and Self-Healing in Wireless Networks," ACM PDC*, 2002.

# VITA

Sung-eok Jeon received the B.S. in Electronics Engineering from Yonsei University, Seoul, Korea, in 1996. He received the M.S. degree in Electronic Engineering from Korea Advanced Institute of Science and Technology, Taejeon, Korea, in 1999. From May 2000 to August 2007, he was a Graduate Research and Teaching Assistant in the Department of Electrical and Computer Engineering at the Georgia Institute of Technology. He completed his Ph.D. in Electrical and Computer Engineering at the Georgia Institute of Technology in August 2007. His research interests are in network management and modeling, with a current focus on randomized distributed management of wireless infrastrueless networks and multi-class networks, distributed and near-optimal resource allocation, scheduling and MAC in wireless networks, and survivability and restoration of optical networks.